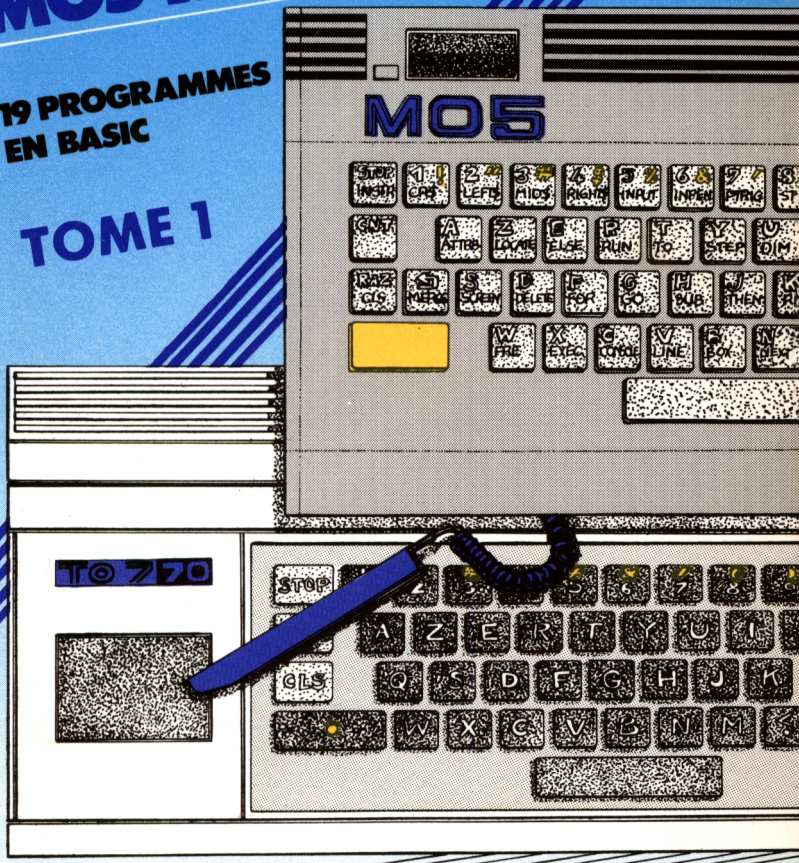
BOITE A OUTILS POUR

MO5 ET TO7/70

MegaD POCHE

19 PROGRAMMES EN BASIC

TOME 1



michel martin

Boîte à outils pour MO5 et TO7/70

Boîte à outils. MO5 et TO7/70 — I.S.B.N. 2.87109.015.7
© s.a. SAPECA, Uccle (Belgique) 1984

Toute reproduction, non strictement réservée à l'usage du copiste, d'un extrait quelconque de ce livre par quelque procédé que ce soit, est interdite sans autorisation écrite de l'éditeur.

Les collections **MégaO** (Poche) sont éditées par SAPECA, 5 avenue de la Ferme Rose, 1180 Bruxelles (Roze Hoevelaan - 1180 Brussel). Le label **MégaO,** les titres des collections et la présentation des volumes sont déposés conformément à la loi. Distributeurs en **France :** PSI DIFFUSION, BP 86, 77402 Lagny-sur-Marne Cedex. Pour le **Canada** et les **USA :** S.C.E. Inc., 65 Avenue Hillside, Westmount - Québec H3 Z1 W1. Pour le **Benelux :** PSI Benelux, 5 Avenue de la Ferme Rose, 1180 Bruxelles.

Boîte à outils pour MO5 et TO7/70

19 programmes en Basic

TOME 1

par
Michel Martin

POCHE

1984

Avertissement

Ce livre s'adresse aux possesseurs de MO5 ou TO7 qui désirent mettre en oeuvre des programmes immédiatement opérationnels.

STRUCTURE DU LIVRE

Chaque programme fait l'objet de cinq parties :

- ***présentation*** : *description macroscopique du programme.*
- ***organigramme*** : *définition de la structure du programme.*
- ***description modulaire du programme*** : *niveaux 1, 2 et 3.*
- ***identification des variables*** : *description fonctionnelle.*
- ***listing du programme.***

COMMENT TROUVER UN PROGRAMME

Les programmes sont classés par thème (voir sommaire).

TYPE DE PROGRAMMATION

Dans la mesure du possible, les programmes sont :

- ***structurés*** : *ils possèdent une base (ou structure) bien définie et logique ;*
- ***hiérarchisés*** : *ils sont composés de modules de niveaux de complexité (ou de hiérarchie) différents.*
  Chaque programme possède généralement trois niveaux hiérarchiques :
  - *niveau 1 : module de commande (niveau hiérarchique élevé).*
  - *niveau 2 : trois modules principaux (début, traitement, fin).*
  - *niveau 3 : procédures élémentaires, souvent sollicitées par les niveaux 1 et 2 (niveau hiérarchique bas).*

Tous les programmes présentés dans ce livre fonctionnent sans aucune modification sur les micro-ordinateurs MO5 et TO7.

Sommaire

JEUX - D'adresse ou de rapidité

- Réflexe 8
- Casse-pipes 12
- Attaque aérienne 18
- Le piéton courageux 25

JEUX - De hasard

- Jeu du 21 34
- Jack-pot 39
- Le cochon qui rit 46

JEUX - De réflexion

- Jeu des allumettes 54
- Mémoire à court terme 59
- Simon 64
- Mastermind 71

GRAPHIQUES ET MATHEMATIQUES

- Figures de Moivre 78
- Formulaire mathématique 82
- Le calendrier perpétuel 89
- Calendrier 93
- Biorythmes 103

MO5/TO7 JOUE ET DESSINE

- Piano 110
- Thèmes célèbres 114
- Titi 119

Annexe 125

JEUX D'ADRESSE OU DE RAPIDITÉ

Réflexe

PRESENTATION

Connaissez-vous la disposition des touches sur le clavier ? Si non, mémorisez-la rapidement grâce à ce programme. Si oui, testez votre réflexe.

MO5/TO7 affiche un caractère sur l'écran. Vous devez taper le même caractère, le plus rapidement possible.

Alors, à vos claviers...

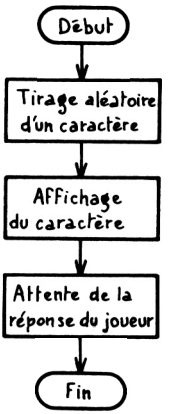
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à deux niveaux définie en annexe.

Le module de traitement est simple. Il consiste en :

- tirage aléatoire d'un caractère alphanumérique ;
- affichage à l'écran de ce caractère ;
- attente de la réponse du joueur.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 90** : Module de commande.

Niveau 2 - **lignes 1000 à 1350** : Présentation du jeu.
lignes 2000 à 2200 : Jeu.
lignes 3000 à 3100 : Fin du jeu.

IDENTIFICATION DES VARIABLES

A : code ASCII du caractère tiré au hasard.

A$: attente d'un caractère pour passer à la suite.

I : index de boucles.

R$: indicateur de répétition de partie.

T : temps de réflexion.

```
10 :    REM REFLEXE
20 :
30 :      GOSUB 1000 'Presentation
40 :      GOSUB 2000 'Jeu
50 :      GOSUB 3000 'Fin du jeu
60 :      IF R$="O" THEN 40 'Nouvelle Partie
70 :
80 :    END
90 :    REM ****************************************
1000 :  REM Presentation du jeu
1010 :
1020 :    SCREEN 6,0,0 : LOCATE 1,1,0
1030 :
1040 :    REM Titre
1050 :      ATTRB 1,1 : COLOR 1,0 : CLS
1060 :      PRINT : PRINT SPC(5)"REFLEXE"
1070 :
1080 :    REM Objet
1090 :
1100 :      ATTRB 0,0 : PRINT:PRINT:PRINT:PRINT
1110 :      COLOR 4,0
1120 :      PRINT"OBJET : ";
1130 :      COLOR 6,0
1140 :      PRINT"Jeu de rapidite."
1150 :
1160 :    REM Job
1170 :
1180 :      PRINT:PRINT:PRINT:PRINT
1190 :      COLOR 5,0
1200 :      PRINT"JOB    : ";
1205 :      COLOR 6,0
1210 :      PRINT"Une lettre va apparaitre sur"
1220 :      PRINT"          l'ecran . Appuyez sur la touc
he"
1230 :      PRINT"          correspondante en un minimum"
1235 :      PRINT"          de temps."
1240 :
1250 :    REM Debut
1260 :
1270 :      PRINT:PRINT:PRINT:PRINT
1280 :      COLOR 3,0
1290 :      PRINT"DEBUT : ";
1295 :      COLOR 6,0
1300 :      PRINT"Pour commencer , appuyez sur"
1310 :      PRINT"          une touche quelconque."
1320 :      A=RND:A$=INKEY$:IF A$="" THEN 1320 'Attente
 de l'appui
```

```
1330 :
1340 : RETURN
1350 : REM ****************************************
2000 : REM Jeu
2010 :
2020 :    REM Tirage aleatoire d'une lettre
2030 :
2040 :      A=INT(RND*26)+65
2050 :
2060 :    REM Affichage de la lettre tiree
2070 :
2080 :      CLS : LOCATE 1,1,0
2090 :      T=0 'RAZ temps de reflexion
2100 :      FOR I=1 TO 11 : PRINT : NEXT I
2110 :      ATTRB 1,1 : COLOR 1,0
2120 :      PRINT SPC(10)CHR$(A)
2130 :
2140 :    REM Attente de l'appui sur une touche
2150 :
2160 :      B$=INKEY$
2170 :      T=T+1 : IF B$="" THEN 2160
2180 :
2190 : RETURN
2200 : REM ****************************************
3000 : REM Fin du jeu, affichage des resultats
3010 :
3020 :    ATTRB 0,0 : CLS
3030 :    IF B$<>CHR$(A) THEN PRINT"Il faudra acheter d
es lunettes !!"
3040 :    PRINT:PRINT
3050 :    PRINT"Temps de reflexion :";T
3060 :    PRINT:PRINT:PRINT
3070 :    COLOR 6,0 : INPUT"Un autre essai (O/N) ";R$
3080 :
3090 : RETURN
3100 : REM ****************************************
```

Casse-pipes

PRESENTATION

Connaissez-vous ce stand de tir présent dans toutes les fêtes foraines, où le joueur doit abattre plusieurs pipes qui se déplacent à vitesse constante ?

Ce programme simule un tel stand de tir. Une seule différence : les pipes sont fixes, et le joueur se déplace à vitesse constante.

Le nombre de balles n'est pas limité. A chaque partie gagnée, la vitesse de déplacement du joueur augmente. Le but du jeu est de détruire toutes les pipes avant de se trouver dans leur proximité immédiate.

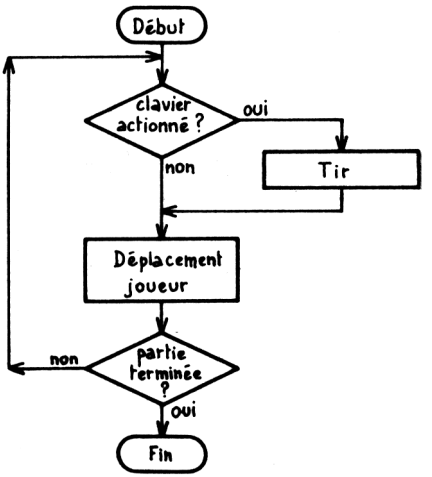
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- test si clavier actionné ;
- si oui, tir joueur ;
- déplacement joueur ;
- test fin de partie.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 90** : Module de commande.

Niveau 2 - **lignes 1000 à 1490** : Présentation du jeu.
lignes 2000 à 2060 : Jeu.
lignes 4000 à 4140 : Fin du jeu.

Niveau 3 - **lignes 2100 à 2480** : Initialisation de la partie.
lignes 3000 à 3270 : Partie.
lignes 3100 à 3180 : - tir.
lignes 3200 à 3270 : - déplacement joueur.

IDENTIFICATION DES VARIABLES

A : lecture de données.

A$: attente d'un caractère pour passer à la suite.

B : nombre de pipes au départ.

FIN : indicateur de répétition de partie.

I,J : index de boucles.

NP : nombre de pipes dans la partie.

PAS : pas de progression verticale du joueur.

PE : numéro de partie.

SC : score joueur.

T : nombre de tirs.

X,Y : position du joueur sur l'écran.

```
10 :   REM CASSE-PIPES
20 :
25 :     CLEAR,,3 'Declaration de Trois motifs 9raPhiq
ues
30 :     GOSUB 1000 'Presentation
40 :     GOSUB 2000 'Initialisation et Jeu
50 :     GOSUB 4000 'Fin du Jeu
60 :     IF FIN=0 THEN 40 'Nouvelle Partie
80 :   END
90 :   REM ****************************************
1000 : REM Presentation du Jeu
1010 :
1020 :   SCREEN 6,0,0 : LOCATE 1,1,0
1030 :
1040 :   REM Titre
1050 :
1060 :     ATTRB 1,1 : COLOR 1,0 : CLS
1070 :     PRINT : PRINT SPC(3)"CASSE - PIPES"
1075 :
1080 :   REM Objet
1090 :
1100 :     ATTRB 0,0 : PRINT:PRINT:PRINT:PRINT
1110 :     COLOR 4,0
1120 :     PRINT"OBJET : ";
1130 :     COLOR 6,0
1140 :     PRINT"Jeu de reflexe."
1150 :
1160 :   REM Job
1170 :
1180 :     PRINT:PRINT:PRINT
1190 :     COLOR 5,0
1200 :     PRINT"JOB    : ";
1210 :     COLOR 6,0
1220 :     PRINT"Le but du jeu est d'abattre"
1230 :     PRINT"         un maximum de PiPes en un tem
Ps"
1240 :     PRINT"         fixe dePendant du talent du"
1250 :     PRINT"         joueur."
```

```
1260 :      PRINT"          La touche ESPACE permet"
1270 :      PRINT"          d'activer le tir."
1280 :
1285 :
1290 :    REM Debut
1300 :
1310 :      PRINT:PRINT:PRINT
1320 :      COLOR 3,0
1330 :      PRINT"DEBUT : ";
1340 :      COLOR 6,0
1350 :      PRINT"Pour commencer, appuyez sur"
1360 :      PRINT"          une touche quelconque."
1370 :
1380 :    B=RND:A$=INKEY$:IF A$="" THEN 1380 'Attente a
ppui
1390 :
1400 :  REM Initialisation des caracteres graphiques
1410 :
1420 :    PAS=1 : DIM T(20)
1430 :
1440 :    DEFGR$(0)=0,0,2,7,255,255,255,0    'Pipe
1450 :    DEFGR$(1)=0,0,170,85,170,85,0,0    'Brique
1460 :    DEFGR$(2)=24,24,24,24,255,255,0,0 'Joueur
1470 :
1480 :  RETURN
1490 :  REM ****************************************
2000 :  REM Initialisation et Jeu
2010 :
2020 :    GOSUB 2100 'Initialisation d'une Partie
2030 :    GOSUB 3000 'Partie
2040 :
2050 :  RETURN
2060 :  REM ****************************************
2100 :  REM Initialisation d'une Partie
2110 :
2120 :    REM Titre
2130 :
2140 :      ATTRB 1,1 : COLOR 1,0 : CLS
2150 :      PRINT : PRINT SPC(3)"CASSE - PIPES"
2160 :
2170 :    REM Tirage aleatoire des Pipes
2180 :
2190 :
2195 :      COLOR 4
2200 :      FOR I=1 TO 37 STEP 2
2210 :        B=INT(RND*3)+1:J=J+1
2220 :        IF B=3 THEN B$=GR$(1):T(J)=0 ELSE B$=GR$(
0):NP=NP+1:T(J)=1
```

```
2230 :         PSET (I,4)B$
2240 :       NEXT I
2250 :
2260 :     REM Curseur en Position
2270 :
2275 :       ATTRB 0,0
2280 :       FOR I=1 TO 18
2290 :         PRINT
2300 :       NEXT I
2310 :
2320 :     REM Commentaires debut de jeu
2330 :
2340 :       COLOR 3
2350 :       PRINT"    Pressez une touche Pour commencer
"
2360 :       A$=INPUT$(1) 'Attente aPPui
2370 :
2380 :     REM Debut de jeu
2390 :
2400 :       COLOR 6
2410 :       PRINT"        Score : "SC"      Tir : "T
2420 :       PRINT CHR$(11)CHR$(11)SPC(40)
2430 :       X=1:Y=20:PSET (X,20)GR$(2)
2440 :
2450 :     PRINT:PRINT:PRINT'Joueur au dePart
2460 :
2470 :   RETURN
2480 :   REM ****************************************
3000 :   REM Partie
3010 :
3020 :     A$=INKEY$
3030 :     IF A$<>"" THEN GOSUB 3100 'Action Joueur
3040 :     GOSUB 3200 'DePlacement Joueur
3050 :
3060 :     IF Y>=PAS+4 AND SC<>NP THEN 3020
3070 :
3080 :   RETURN
3090 :   REM ****************************************
3100 :   REM Action Joueur
3110 :
3120 :     PLAY"DO":XP=(X+1)/2
3130 :     IF T(XP)=1 THEN T(XP)=0:SC=SC+1:PSET(X,4)" ":
PLAY"MI"
3140 :     T=T+1 'Nombre de tirs
3150 :     LOCATE 1,23:PRINT "       Score : "SC"      Tir
 : "T
3160 :     PRINT CHR$(11);
3170 :
```

```
3180 : RETURN
3190 : REM ****************************************
3200 : REM Deplacement du Joueur
3210 :
3220 :   PSET(X,Y)" "
3230 :   IF X=37 THEN Y=Y-PAS:X=1 ELSE X=X+2
3240 :   PSET(X,Y)GR$(2)
3250 :
3260 : RETURN
3270 : REM ****************************************
4000 : REM Fin de la Partie
4010 :
4020 :   CLS
4030 :   IF SC=NP THEN 4050 'Partie sans faute
4035 :   IF PAS>1 THEN PAS=PAS-1
4040 :   PRINT"Vous n'etes pas en forme dirait-on !"
4045 :   GOTO 4060
4050 :   PRINT"Pas mal, mais peut faire mieux ..."
4055 :   PAS=PAS+1
4060 :   PRINT:PRINT"Score : "SC" Nombre de tirs :"T
4070 :   PRINT"Nombre de PiPes : "NP
4080 :
4090 :   PRINT:INPUT"Une autre Partie(O/N): ";R$
4100 :   IF R$="O" THEN FIN=0 ELSE FIN=1
4105 :
4110 :   NP=0 : SC=0 : T=0 : J=0
4120 :
4130 : RETURN
4140 : REM ****************************************
```

Attaque aérienne

PRESENTATION

Un tank disposé au sol doit faire face à une attaque aérienne. Pour cela, le tank peut se déplacer vers la droite, vers la gauche, et tirer autant de missiles anti-aériens qu'il le désire.

Le jeu s'arrête quand le tank est détruit.

Combien de temps tiendrez-vous face à l'attaque aérienne ?

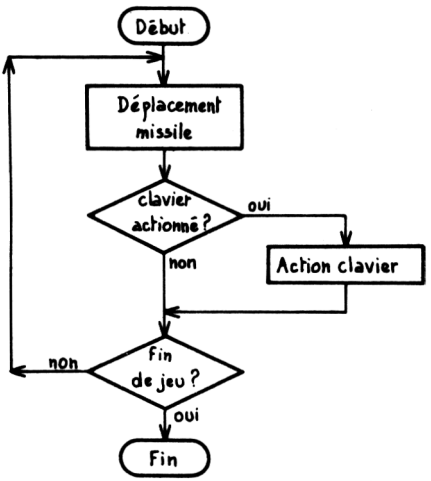
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- déplacement missile ;
- test si clavier actionné ;
- test si fin de jeu.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 70** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Initialisation.
lignes 2000 à 2250 : Jeu.
lignes 3000 à 3060 : Fin du jeu.

Niveau 3 - **lignes 1100 à 1530** : Affichage du menu.
lignes 1700 à 1920 : Initialisation des variables.
lignes 2250 à 2310 : Joueur touché.
lignes 2320 à 2360 : Abri touché.
lignes 2600 à 2660 : Action joueur.
lignes 2700 à 2780 : - tir.
lignes 2800 à 2850 : - déplacement vers la droite.
lignes 2900 à 2950 : - déplacement vers la gauche.

IDENTIFICIATION DES VARIABLES

A : position du missile au départ.

A$: attente d'un caractère pour passer à la suite.

B : déplacement horizontal du missile.

BIS : indicateur de nouveau missile.

B$: caractère déplacement vers le haut.

C : position écran.

CC : position du tank.

CO : colonne missile.

FIN : indicateur de fin de partie.

LI : position ligne du missile.

NB : nombre de missiles touchés.

TC : indicateur de missile touché.

```
10 :    REM ATTAQUE AERIENNE
20 :
25 :      CLEAR,,1 ' Declaration d'un motif 9raPhi9ue
30 :      GOSUB 1000 'Initialisation
40 :      GOSUB 2000 'Jeu
50 :      GOSUB 3000 'Fin du jeu
60 :
65 :    END
70 :    REM ****************************************
1000 :  REM Initialisation du jeu
1010 :
1020 :    GOSUB 1100 'Affiche9e du menu
1030 :    GOSUB 1700 'Initialisation des variables
1040 :
1050 :  RETURN
1060 :  REM ****************************************
1100 :  REM Affiche9e du menu
1110 :
1120 :    SCREEN 6,0,0 : LOCATE 1,1,0
1130 :
1140 :    REM Titre
1150 :
1160 :      ATTRB 1,1 : COLOR 1,0 : CLS
1170 :      PRINT:PRINT"  ATTAQUE AERIENNE"
1180 :
1190 :    REM Objet
1200 :
1210 :      ATTRB 0,0 : PRINT:PRINT
1220 :      COLOR 4
1230 :      PRINT"OBJET : ";
1240 :      COLOR 6
1250 :      PRINT"Jeu d'adresse et de raPidite."
```

```
1260 :
1270 :    REM Job
1280 :
1290 :       PRINT:PRINT
1300 :       COLOR 5
1310 :       PRINT"JOB    : ";
1320 :       COLOR 6
1330 :       PRINT"Vous etes attaque par des"
1340 :       PRINT"          missiles aeriens repetes par
*"
1350 :       PRINT"          Le but du jeu est de les"
1360 :       PRINT"          detruire. Vous disposez de"
1370 :       PRINT"          roquettes envoyees par la"
1380 :       PRINT"          touche ESPACE. Lestouches"
1390 :       PRINT"          deplacement vers la droite et
"
1400 :       PRINT"          vers la gauche gerent votre"
1410 :       PRINT"          position au sol. Vous disPose
z"
1420 :       PRINT"          egalement de 2 abris au sol."
1430 :       PRINT"          Combien d'ennemis tuerez-vous
 ?"
1440 :
1450 :    REM Debut
1460 :
1470 :       PRINT:PRINT
1480 :       COLOR 3
1490 :       PRINT"DEBUT  : ";
1500 :       COLOR 6
1505 :       PRINT"Pour commencer, appuyez sur"
1510 :       PRINT"          une touche quelconque."
1515 :       A=RND:A$=INKEY$:IF A$="" THEN 1515 'Attente
 de l'appui
1520 :
1525 :  RETURN
1530 :  REM ****************************************
1540 :
1700 :  REM Initialisation des variables
1710 :
1720 :    BIS=1 'Missile a l'origine
1730 :    CC=11 'Joueur a gauche de l'ecran
1740 :    CLS 'Effacement de l'ecran
1750 :    FOR I=1 TO 4 'Abris
1751 :       PSET(12+I,18)"="
1752 :       PSET(21+I,18)"="
1753 :    NEXT
1760 :
1770 :    FOR I=1 TO 21 'Cadre horizontal
```

```
1771 :      PSET(8+I,7)"-"
1772 :      PSET(8+I,21)"-"
1773 :    NEXT I
1780 :
1790 :    FOR I=1 TO 13 'Cadre vertical
1791 :      PSET(9,7+I)"I"
1792 :      PSET(29,7+I)"I"
1793 :    NEXT I
1800 :
1810 :    REM Definition du Joueur
1820 :
1830 :      DEFGR$(0)=0,0,24,24,24,24,255,255
1840 :
1850 :    REM Affichage du Joueur
1860 :
1870 :      PSET(11,20)GR$(0):PJ=11
1880 :
1890 :    LOCATE 1,2:PRINT"Score : 0"
1900 :
1910 :  RETURN
1920 :  REM ****************************************
2000 :  REM Jeu
2010 :
2020 :    GOSUB 2100 'Deplacement missile
2030 :    A$=INKEY$:IF A$<>"" THEN GOSUB 2600 'Action J
oueur
2040 :    IF FIN=0 THEN GOTO 2020
2045 :
2050 :  RETURN
2060 :  REM ****************************************
2100 :  REM Deplacement missile
2110 :
2120 :    IF BIS<>1 THEN 2160
2130 :    A=INT(RND*3):A=A*2+17 'Missile au depart
2140 :    B=INT(RND*3)-1 'Deplacement horizontal
2150 :    PSET(A,8)"*":LI=10:CO=A+B:BIS=0:RETURN
2160 :    C=SCREEN(CO,LI) 'Nouvelle Position missile
2170 :    PSET(CO-B,LI-2)" "
2180 :    IF LI=22 THEN BIS=1:RETURN
2190 :    IF LI=20 AND CO=PJ THEN 2260 'Joueur touche
2200 :    IF C=61 THEN 2320 'Abri touche
2210 :    PSET(CO,LI)"*" 'Avance missile
2220 :    LI=LI+2:CO=CO+B 'Prochaine Position missile
2230 :
2240 :  RETURN
2250 :  REM ****************************************
2260 :  REM Joueur touche
2270 :
```

```
2280 :    PLAY"L2005MIMIMI":FIN=1
2290 :
2300 : RETURN
2310 : REM ****************************************
2320 :  REM Abri touche
2330 :
2340 :     PLAY"MI":PSET(CO-1,LI)" ":PSET(CO,LI)" ":BIS
=1
2345 :
2350 : RETURN
2360 : REM ****************************************
2600 : REM Action Joueur
2610 :
2620 :    IF A$=" " THEN 2700 'Tir Joueur
2630 :    IF ASC(A$)=9 THEN 2800 'Depl. ->
2640 :    IF ASC(A$)=8 THEN 2900 'Depl. <-
2645 :
2650 : RETURN
2660 : REM ****************************************
2700 : REM Tir Joueur
2710 :
2720 :    FOR I=8 TO 18 STEP 2
2730 :       IF SCREEN(CC,I)=42 THEN TC=1:J=I:I=18:PSET(
CC,J)"O"
2740 :    NEXT I
2750 :    IF TC=1 THEN PLAY"SO":NB=NB+1:BIS=1:PSET(CC,J
)" "
2755 :    IF TC=1 THEN LOCATE 8,2:PRINT NB:TC=0
2760 :    IF NB=50 THEN FIN=1 'Jeu gagne
2765 :
2770 : RETURN
2780 : REM ****************************************
2800 : REM Deplacement vers la droite
2810 :
2820 :    IF CC=27 THEN RETURN
2830 :    PSET(CC,20)" ":CC=CC+2:PJ=PJ+2:PSET(CC,20)GR$
(0)
2835 :
2840 : RETURN
2850 : REM ****************************************
2900 : REM Deplacement vers la gauche
2910 :
2920 :    IF CC=11 THEN RETURN
2930 :    PSET(CC,20)" ":CC=CC-2:PJ=PJ-2:PSET(CC,20)GR$
(0)
2935 :
2940 : RETURN
2950 : REM ****************************************
```

```
3000 : REM Fin de Partie
3010 :
3020 :    CLS
3030 :    IF NB=50 THEN PRINT"Attaque refoulee. Bravo !
!"
3031 :    IF NB<>50 THEN PRINT"Tu n'es plus de ce monde
!!"
3040 :
3050 : RETURN
3060 : REM ****************************************
```

Le piéton courageux

PRESENTATION

Vous traversez la route tous les jours, et cela est devenu une habitude pour vous. Ce programme se propose de réactualiser vos préjugés. Vous jouez le rôle d'un piéton courageux qui veut traverser une autoroute à quatre voies, le plus souvent possible dans un temps maximum de une minute. La partie s'arrête après écoulement du temps imparti, ou si le piéton se fait écraser.

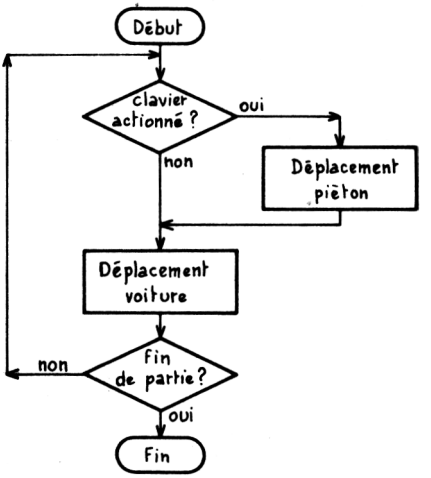
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- test si clavier actionné ;
- déplacement voiture ;
- test si fin de partie.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1080** : Initialisation du jeu.
lignes 2000 à 2060 : Jeu.
lignes 3000 à 3060 : Fin du jeu.

Niveau 3 - **lignes 1100 à 1430** : Présentation du jeu.
lignes 1600 à 1680 : Définition des caractères graphiques.
lignes 1700 à 1780 : Définition des constantes et variables.
lignes 1800 à 1950 : Dessin de la grille du jeu.
lignes 2070 à 2170 : Déplacement du piéton.
lignes 2200 à 2290 : - vers le haut.
lignes 2300 à 2370 : - vers le bas.
lignes 2400 à 2610 : Déplacement voiture.

IDENTIFICATION DES VARIABLES

A$: attente d'un caractère pour passer à la suite.

FIN : indicateur de fin de partie.

I,J : index de boucles.

PJ : position du joueur.

T : tableau indiquant la position des voitures.

VO : numéro de voiture déplacée (1, 2, 3 ou 4).

X : nombre de traversées piéton.

```
10 :   REM LE PIETON COURAGEUX
20 :
25 :     CLEAR,,2 'Declaration de deux motifs 9raPhiqu
es
30 :     GOSUB 1000 'Initialisation
40 :     GOSUB 2000 'Jeu
50 :     GOSUB 3000 'Fin du jeu
60 :
70 :   END
80 :   REM ****************************************
1000 : REM Initialisation
1010 :
1020 :   GOSUB 1100 'Presentation
1030 :   GOSUB 1600 'Definition des caracteres
1040 :   GOSUB 1700 'Definition des constantes et vari
ables
1050 :   GOSUB 1800 'Dessin de la 9rille de jeu
1060 :
1070 : RETURN
1080 : REM ****************************************
1100 : REM Presentation du jeu
1110 :
1120 :   SCREEN 6,0,0 : LOCATE 1,1,0
1130 :
1140 :   REM Titre
1150 :
1160 :     ATTRB 1,1 : COLOR 1,0 : CLS
1170 :     PRINT:PRINT"LE PIETON COURAGEUX"
1180 :
1190 :   REM Objet
1200 :
1210 :     ATTRB 0,0 : PRINT:PRINT:PRINT
1220 :     COLOR 4
1230 :     PRINT"OBJET : ";
1240 :     COLOR 6
1250 :     PRINT"Jeu d'adresse."
1260 :
1270 :   REM Job
1280 :
```

```
1290 :      PRINT:PRINT:PRINT
1300 :      COLOR 5
1310 :      PRINT"JOB    : ";
1320 :      COLOR 6
1330 :      PRINT"Le but du jeu est de"
1340 :      PRINT"         traverser la route le Plus de
"
1350 :      PRINT"         fois Possible en se servant"
1360 :      PRINT"         des touches-fleches vers le b
as"
1370 :      PRINT"         et vers le haut."
1380 :      PRINT"         Le temPs est limite"
1390 :      PRINT"         a une minute."
1400 :
1410 :    REM Debut
1420 :
1421 :      PRINT:PRINT:PRINT
1422 :      COLOR 3
1423 :      PRINT"DEBUT : ";
1424 :      COLOR 6
1425 :      PRINT"Pour commencer, aPPuyez sur"
1426 :      PRINT"         une touche 9uelcon9ue."
1427 :      A=RND:A$=INKEY$:IF A$="" THEN 1427 'Attente
 de l'aPPui
1428 :
1429 : RETURN
1430 : REM ************************
1600 : REM Definition des caracteres 9raPhi9ues
1610 :
1620 :    REM Voiture
1630 :    DEFGR$(0)=0,28,28,8,28,8,8,20
1640 :
1650 :    REM Pieton
1660 :    DEFGR$(1)=0,0,12,63,63,18,0,0
1665 :
1670 : RETURN
1680 : REM ****************************************
1700 : REM Definition des constantes et variables du j
eu
1710 :
1720 :    V0=1 'No de voie Pour voiture
1730 :    T(1)=9:T(2)=9 'Voies 1 et 2, voitures a droit
e
1740 :    T(3)=1:T(4)=1 'Voies 3 et 4, voitures a 9auch
e
1750 :    PJ=0 'Position du Pieton
1760 :
1770 : RETURN
1780 : REM ****************************************
```

```
1800 : REM Dessin de la grille du jeu
1810 :
1815 :   CLS:PRINT"SCORE : 0"
1820 :   FOR I=1 TO 5
1830 :     FOR J=1 TO 19
1840 :       PSET(10+J,9+2*I)"-"
1850 :     NEXT J
1860 :   NEXT I
1870 :   FOR I=1 TO 19
1880 :     PSET(10+I,15)"="
1890 :   NEXT I
1900 :   PSET(28,12)GR$(1):PSET(28,14)GR$(1) 'Voitures
 1 et 2
1910 :   PSET(12,16)GR$(1):PSET(12,18)GR$(1) 'Voitures
 3 et 4
1920 :   PSET(20,20)GR$(0) 'Pieton
1930 :
1940 : RETURN
1950 : REM ****************************************
2000 : REM Deroulement du jeu
2010 :
2020 :   A$=INKEY$:IF A$<>"" THEN GOSUB 2100 'Action J
oueur
2030 :   GOSUB 2400 'Avance voitures
2040 :   IF FIN=0 THEN 2020
2050 :
2060 : RETURN
2070 : REM ****************************************
2100 : REM Deplacement du Pieton
2110 :
2120 :
2130 :   IF ASC(A$)=11 THEN 2200 'Vers le haut
2140 :   IF ASC(A$)=10 THEN 2300 'Vers le bas
2150 :
2160 : RETURN
2170 : REM ****************************************
2200 : REM Deplacement vers le haut
2210 :
2220 :   PJ=PJ+1
2230 :   FOR I=1 TO 4
2231 :     IF PJ=5-I AND T(I)=5 THEN FIN=2 'Collision
2232 :   NEXT I
2240 :   IF PJ<>5 THEN 2270
2250 :   PSET(20,12)" ":PJ=0:SC=SC+1
2260 :   PSET(20,20)GR$(0):PLAY"MI":X=X+1:LOCATE 7,0:P
RINT X:RETURN
2270 :   PSET(20,20-2*PJ)GR$(0):PSET(20,22-PJ*2)" "
2275 :
2280 : RETURN
```

```
2290 : REM ****************************************
2300 : REM DePlacement vers le bas
2310 :
2320 :    PJ=PJ-1
2330 :    FOR I=1 TO 4
2331 :       IF PJ=5-I AND T(I)=5 THEN FIN=3 'Collision
2332 :    NEXT I
2340 :    IF PJ=0 THEN PJ=1:RETURN
2350 :    IF PJ=-1 THEN PJ=0
2355 :    PSET(20,20-2*PJ)GR$(0):PSET(20,18-2*PJ)" "
2360 :
2365 : RETURN
2370 : REM ****************************************
2400 : REM DePlacement des voitures
2410 :
2420 :    TE=TE+1:IF TE=460 THEN FIN=4 'TemPs ecoule
2425 :    IF VO=1 OR VO=2 THEN 2520 'DePlacement <-
2426 :
2430 :    REM DePlacement vers la droite
2431 :
2440 :       IF T(VO)<>9 THEN 2480
2450 :       PSET(28,10+2*VO)" ":PSET(12,10+2*VO)GR$(1):
T(VO)=1
2460 :       VO=INT(RND*4)+1
2470 :       RETURN
2480 :       IF T(VO)=4 AND PJ=5-VO THEN FIN=5 'Collisio
n
2490 :       PSET(10+T(VO)*2,10+VO*2)" ":T(VO)=T(VO)+1
2495 :       PSET(10+T(VO)*2,10+VO*2)GR$(1)
2500 :       VO=INT(RND*4)+1
2505 :
2510 :    RETURN
2515 :
2520 :    REM DePlacement vers la Gauche
2521 :
2530 :       IF T(VO)<>1 THEN 2560
2540 :       PSET(12,10+VO*2)" ":T(VO)=9:PSET(28,10+VO*2
)GR$(1)
2550 :       VO=INT(RND*4)+1
2555 :       RETURN
2560 :       IF T(VO)=6 AND PJ=5-VO THEN FIN=6 'Collisio
n
2570 :       PSET(10+T(VO)*2,10+VO*2)" ":T(VO)=T(VO)-1
2580 :       PSET(10+T(VO)*2,10+VO*2)GR$(1)
2590 :       VO=INT(RND*4)+1
2595 :
2600 : RETURN
2610 : REM ****************************************
```

```
3000 : REM Fin de Partie
3010 :
3020 :   CLS:PLAY"T4LALALA"
3030 :   IF FIN>=5 THEN PRINT"Ecrase !!!"
3035 :   IF FIN=4 THEN PRINT"Temps ecoule."X"traversee
(s)."
3040 :   IF FIN<4 THEN PRINT"On se jete sur les voitur
es maintenant ?"
3050 :
3060 : RETURN
```

JEUX DE HASARD

Jeu du 21

PRESENTATION

Ce jeu vous oppose à MO5/TO7. Par tir de dés répétés, vous, puis l'ordinateur devez vous rapprocher au maximum du score 21, sans jamais le dépasser. Si votre score est plus proche de 21 que celui de MO5/TO7, vous doublez votre mise, sinon, vous la perdez.

Saurez-vous vous arrêter à temps ?

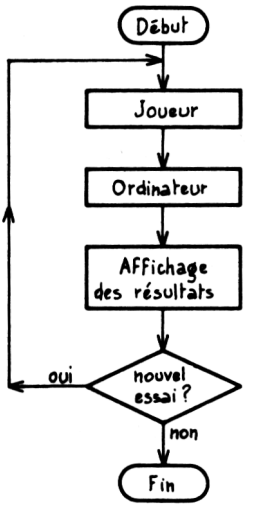
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement est simple. Il consiste en :

- tirages de dés par joueur ;
- tirages de dés par ordinateur ;
- affichage des résultats ;
- éventuellement, nouvel essai.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1400** : Initialisation.
lignes 2000 à 2070 : Jeu.
lignes 3000 à 3040 : Fin du jeu.

Niveau 3 - **lignes 2100 à 2240** : Jeu utilisateur.
lignes 2300 à 2360 : Jeu ordinateur.
lignes 2400 à 2440 : Tirage aléatoire.
lignes 2500 à 2630 : Affichage des résultats.

IDENTIFICATION DES VARIABLES

A : dé.

AR : argent disponible pour le joueur.

A$: caractère action joueur.

MISE : argent misé par le joueur.

MO5 : total MO5/TO7.

PERDU : variable de test de fin de partie.

USER : total utilisateur.

```
10 :    REM JEU DU 21
20 :
30 :      GOSUB 1000 'Presentation
40 :      GOSUB 2000 'Jeu
50 :      GOSUB 3000 'Fin du jeu
60 :
70 :    END
80 :    REM ****************************************
1000 :  REM Presentation
1010 :
1020 :    SCREEN 6,0,0 : LOCATE 1,1,0
1030 :
1040 :    REM Titre
1050 :
1060 :      ATTRB 1,1 : COLOR 1,0 : CLS
1070 :      PRINT : PRINT SPC(5)"JEU DU 21"
1080 :
1090 :    REM Objet
1100 :
1110 :      ATTRB 0,0 : PRINT:PRINT:PRINT
1120 :      COLOR 4
1130 :      PRINT"OBJET : ";
1140 :      COLOR 6
1150 :      PRINT"Jeu de hasard."
1160 :
1170 :    REM Job
1180 :
1190 :      PRINT:PRINT:PRINT
1200 :      COLOR 5
1210 :      PRINT"JOB    : ";
1220 :      COLOR 6
1230 :      PRINT"Vous Possedez 100 Francs."
1240 :      PRINT"          Misez une somme inferieure a"
1250 :      PRINT"          100 Francs. Lancez le de"
1260 :      PRINT"          autant de fois que vous le "
1270 :      PRINT"          voulez Pour vous raPProcher"
1280 :      PRINT"          au maximum du score 21."
1290 :      PRINT"          MO5 se mesurera alors a vous.
"
1300 :
1310 :    REM Debut
1320 :
```

```
1330 :      PRINT:PRINT:PRINT
1340 :      COLOR 3
1350 :      PRINT"DEBUT : ";
1360 :      COLOR 6
1370 :      PRINT"Pour commencer, aPPuyez sur"
1380 :      PRINT"         une touche quelconque."
1385 :      A=RND:A$=INKEY$:IF A$="" THEN 1385 'Attente
 de l'aPPui
1390 :
1391 :    REM Somme disPonible au dePart
1392 :
1393 :      AR=100
1394 :
1395 : RETURN
1400 : REM ****************************************
2000 : REM Jeu
2010 :
2015 :    MO5=0:USER=0
2020 :    GOSUB 2100 'Jeu utilisateur
2030 :    GOSUB 2300 'Jeu MO5/TO7
2035 :    GOSUB 2500 'Affichage des resultats
2040 :    INPUT"Nouvel essai (O/N) ";R$
2050 :    IF R$="O" THEN 2000 'Nouvel essai
2055 :
2060 : RETURN
2070 : REM ****************************************
2100 : REM Jeu de l'utilisateur
2110 :
2120 :    CLS
2130 :    INPUT"Mise ";MISE
2140 :    PRINT:PRINT:PRINT"Pour obtenir un nouveau tir
, aPPuyez"
2150 :    PRINT"sur T. Pour faire jouer MO5,"
2160 :    PRINT"aPPuyez sur F."
2170 :    A$=INPUT$(1):IF A$<>"T" THEN 2220
2180 :    GOSUB 2400 'Tir du de
2190 :    USER=USER+A
2200 :    IF USER>21 THEN PERDU=1
2210 :    PRINT"Tir : ";A;"    Total : "USER
2220 :    IF A$="F" OR USER>21 THEN RETURN
2225 :    GOTO 2170
2230 :
2240 : REM ****************************************
2300 : REM Jeu MO5/TO7
2310 :
2320 :    GOSUB 2400 'Tirage aleatoire
2330 :    MO5=MO5+A
2340 :    IF MO5>19 THEN RETURN ELSE 2300
```

```
2350 :
2360 : REM ****************************************
2400 : REM Tirage aleatoire d'une variable A
2410 :
2420 :    A=INT(RND*6)+1
2430 :
2440 : RETURN
2450 : REM ****************************************
2500 : REM Affichage des resultats
2510 :
2520 :    PRINT"Votre Score : "USER
2525 :
2530 :    PRINT"Votre Mise  : "MISE
2540 :    IF PERDU<>1 THEN 2590
2550 :    PRINT"Perdu : Vous avez depasse 21 !"
2560 :    AR=AR-MISE
2570 :    PRINT"Vous Possedez : "AR
2580 :    PERDU=0 : RETURN
2590 :    PRINT"Score MOI : "MOI
2600 :    IF MOI<USER OR MOI>21 THEN PRINT"Bien joue":A
R=AR+MISE
2610 :    IF MOI>USER AND MOI<=21 THEN AR=AR-MISE:PRINT
"Perdu.Il reste"AR
2620 :    IF MOI=USER THEN PRINT"Egalite !!"
2626 : RETURN
2630 : REM ****************************************
3000 : REM Fin du jeu
3010 :
3020 :    COLOR 3:PRINT:PRINT"A la prochaine ...":COLOR
 6
3030 :
3040 : REM ****************************************
```

Jack-pot

PRESENTATION

Ce programme simule le très célèbre jeu de café du même nom. Trois roulettes représentant des fruits (poire, banane, pomme, cerise) sont animées d'un mouvement de rotation. Lorsqu'elles s'arrêtent :

- si les trois roulettes représentent des fruits différents, la mise est perdue ;
- si deux des roulettes représentent le même fruit, vous faites un double. Gain : 10 $;
- si les trois roulettes représentent le même fruit, vous faites un Jack Pot. Gain : 30 $.

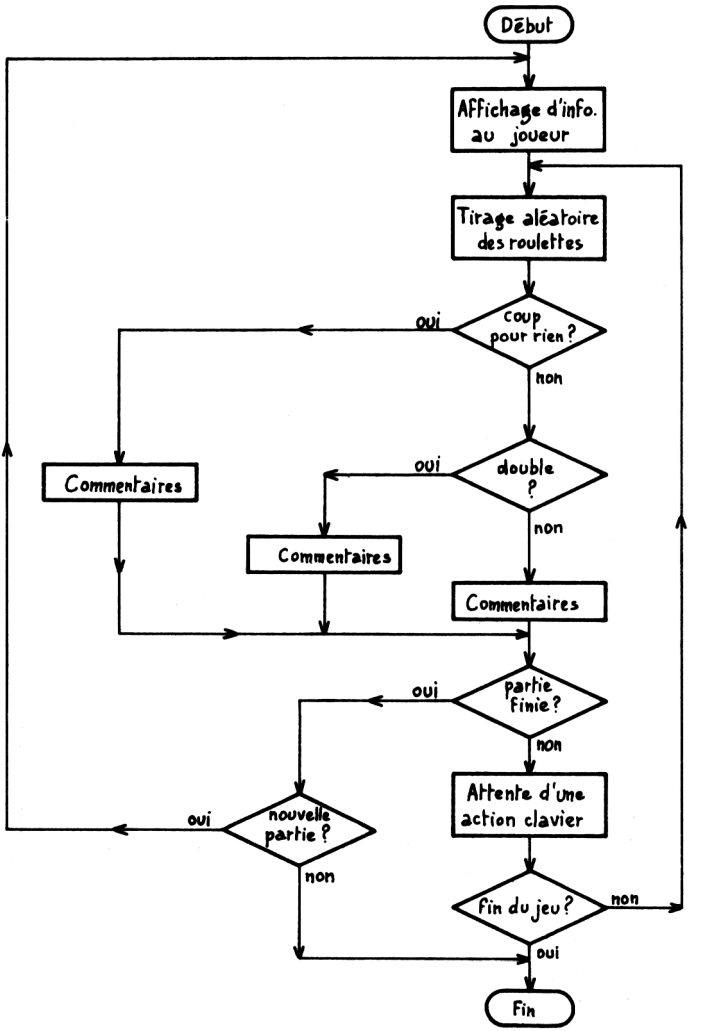
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- affichage d'informations au joueur ;
- tirage aléatoire des roulettes, et commentaires ;
- test de fin de partie et/ou de fin de jeu.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 90** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Présentation et initialisation.
lignes 2000 à 2850 : Jack pot.
lignes 4000 à 4090 : Fin de partie.

Niveau 3 - **lignes 1100 à 1510** : Présentation du jeu.
lignes 1700 à 1960 : Initialisation des caractères graphiques.
lignes 2900 à 2960 : Coup pour rien.
lignes 3000 à 3060 : Double.
lignes 3100 à 3160 : Jack pot.

IDENTIFICATION DES VARIABLES

A$: attente d'un caractère pour passer à la suite.

C1, C2, C3 : fruit sélectionné sur 1ère, 2e, 3e roulette.

E : fruit à afficher.

FIN : indicateur de fin de partie.

I : index de boucles.

INFO$: commentaires sur un coup.

R$: indicateur de répétition de partie.

S : somme disponible.

T$: tableau des fruits.

```
10 :    REM JACK POT
20 :
25 :      CLEAR,,4 'Declaration de 4 motifs graphiques
30 :      GOSUB 1000 'Presentation et Initialisation
40 :      GOSUB 2000 'Jeu
50 :      GOSUB 4000 'Fin de la Partie
60 :      IF R$="O" THEN 40 'Nouvelle Partie
70 :
80 :    END
90 :    REM ****************************************
1000 :  REM Presentation et Initialisation
1010 :
1020 :    GOSUB 1100 'Presentation
1030 :    GOSUB 1700 'Initialisation
1040 :
1050 :  RETURN
1060 :  REM ****************************************
1100 :  REM Presentation du jeu
1110 :
```

```
1120 :    SCREEN 6,0,0 : LOCATE 1,1,0
1130 :
1140 :    REM Titre
1150 :
1160 :      ATTRB 1,1 : COLOR 1,0 : CLS
1170 :      PRINT:PRINT SPC(5)"JACK POT"
1180 :
1190 :    REM Objet
1200 :
1210 :      ATTRB 0,0 : PRINT:PRINT
1220 :      COLOR 4
1230 :      PRINT"OBJET : ";
1240 :      COLOR 6
1250 :      PRINT"Jeu de hasard."
1260 :
1270 :    REM Job
1280 :
1290 :      PRINT:PRINT
1300 :      COLOR 5
1310 :      PRINT"JOB    : ";
1320 :      COLOR 6
1330 :      PRINT"Le jeu consiste a lancer 3"
1340 :      PRINT"         roulettes comportant chacune:
"
1350 :      PRINT:PRINT"              - Une Poire,"
1360 :      PRINT"            - Une banane,"
1370 :      PRINT"            - Une Pomme,"
1380 :      PRINT"            - Une cerise,"
1390 :      PRINT:PRINT"        et a obtenir les memes
fruits"
1395 :      PRINT"       sur chaque roulette a l'arret
."
1400 :
1410 :    REM Debut
1420 :
1430 :      PRINT:PRINT
1440 :      COLOR 3
1450 :      PRINT"DEBUT : ";
1460 :      COLOR 6
1470 :      PRINT"Pour commencer, appuyez sur"
1480 :      PRINT"       une touche quelconque."
1490 :      A=RND:A$=INKEY$:IF A$="" THEN 1490 'Attente
 de l'appui
1500 :
1510 :  RETURN
1520 :  REM ****************************************
1700 :  REM Initialisation des caracteres du jeu
1710 :
1720 :    REM Banane
1730 :
```

```
1740 :      DEFGR$(0)=0,0,0,0,51,63,30,0
1750 :
1760 :
1770 :    REM Poire
1780 :
1790 :      DEFGR$(1)=4,4,12,30,30,63,63,30
1800 :
1810 :
1820 :    REM Pomme
1830 :
1840 :      DEFGR$(2)=4,4,30,63,63,63,30,0
1850 :
1860 :
1870 :    REM Cerise
1880 :
1890 :      DEFGR$(3)=0,1,2,4,24,60,60,24
1900 :
1910 :
1950 : RETURN
1960 : REM ****************************************
2000 : REM Jeu
2010 :
2020 :    CLS : S=100
2030 :
2040 :    REM Titre
2050 :
2060 :      ATTRB 1,1 : COLOR 1
2070 :      PRINT:PRINT SPC(5)"JACK POT"
2080 :
2090 :    REM Configuration initiale
2100 :
2105 :      ATTRB 0,0 : COLOR 6
2110 :      LOCATE 10,12:PRINT"----------------"
2120 :      LOCATE 10,13:PRINT"I              I"
2130 :      LOCATE 10,14:PRINT"I              I"
2140 :      LOCATE 10,15:PRINT"I              I"
2150 :      LOCATE 10,16:PRINT"----------------"
2160 :      T$(1)=CHR$(27)+"C"+GR$(0)
2170 :      T$(2)=CHR$(27)+"B"+GR$(1)
2180 :      T$(3)=CHR$(27)+"A"+GR$(2)
2190 :      T$(4)=CHR$(27)+"A"+GR$(3)
2195 :      E$=CHR$(27)+"F":F$="     "
2200 :
2210 :     REM Roulettes au depart
2220 :
2230 :      LOCATE 10,14
2240 :      PRINT"I  "T$(1)F$T$(2)F$T$(3)E$
2250 :
2260 :     REM Information au joueur
2270 :
```

```
2280 :      LOCATE 1,20
2290 :      PRINT"Pour lancer les roulettes,aPPuyez sur
"
2300 :      PRINT"une touche 9uelcon9ue."
2310 :
2320 :      PRINT" Pour arreter le jeu, aPPuyez sur ENT
REE"
2325 :       LOCATE 12,7:COLOR 3:PRINT"Score : 100"
2330 :
2340 :     REM Jeu
2350 :
2360 :      A$=INPUT$(1)
2370 :      IF ASC(A$)=13 THEN FIN=1
2380 :
2390 :     REM Tira9e 1e case
2400 :
2410 :      FOR I=1 TO 20
2420 :        E=INT(RND*4)+1
2430 :        LOCATE 13,14
2440 :        PRINT T$(E)
2450 :        FOR J=1 TO 30:NEXT J
2460 :      NEXT I
2470 :      F$=T$(E) : C1=E 'Memorisation
2480 :
2490 :     REM Tira9e 2e case
2500 :
2510 :      FOR I=1 TO 20
2520 :        E=INT(RND*4)+1
2530 :        LOCATE 18,14
2540 :        PRINT T$(E)
2550 :        FOR J=1 TO 30:NEXT J
2560 :      NEXT I
2570 :      F$=F$+T$(E) : C2=E 'Memorisation
2580 :
2590 :     REM Tira9e 3e case
2600 :
2610 :      FOR I=1 TO 20
2620 :        E=INT(RND*4)+1
2630 :        LOCATE 23,14
2640 :        PRINT T$(E)
2650 :        FOR J=1 TO 30:NEXT J
2660 :      NEXT I
2670 :      F$=F$+T$(E) : C3=E 'Memorisation
2700 :
2710 :     REM Fin du tira9e aleatoire
2720 :     REM ComPtabilisation des 9ains
2730 :
2740 :      IF C1<>C2 AND C1<>C3 THEN D=1 'CouP Pour ri
en
```

```
2750 :      IF C1=C2 OR C1=C3 OR C2=C3 THEN D=2 'Double
2760 :      IF C1=C2 AND C1=C3 THEN D=3 'Jack Pot
2770 :
2780 :      ON D GOSUB 2900,3000,3100
2790 :
2800 :      LOCATE 12,6:COLOR 3
2810 :      PRINT CHR$(24)INFO$
2815 :      LOCATE 12,7
2820 :      PRINTCHR$(24)"Score :";S
2825 :
2830 :      IF S<0 OR FIN=1 THEN RETURN ELSE 2340 'Bouc
le de jeu
2840 :
2850 : REM ****************************************
2900 : REM Coup Pour rien
2910 :
2920 :    INFO$="Coup Pour rien"
2930 :    S=S-10
2940 :
2950 : RETURN
2960 : REM ****************************************
3000 : REM Double
3010 :
3020 :    INFO$="Double !"
3030 :    S=S+10
3040 :
3050 : RETURN
3060 : REM ****************************************
3100 : REM Jack Pot
3110 :
3120 :    INFO$="Jack Pot !!!"
3130 :    S=S+30
3140 :
3150 : RETURN
3160 : REM ****************************************
4000 : REM Fin de Partie
4010 :
4020 :    CLS : COLOR 6
4030 :    PRINT:PRINT
4040 :    PRINT"La Partie est finie."
4045 :    PRINT"Score final :"S
4050 :    PRINT : PRINT
4060 :    INPUT"Une autre Partie";R$
4070 :    FIN=0 : S=100 'RAZ Parametres de depart
4075 :    LOCATE 1,6,0 'Le curseur disParait
4080 :
4090 : RETURN
```

Le cochon qui rit

PRESENTATION

Ce jeu de hasard consiste à reconstituer un cochon auquel il manque les pattes, les yeux, les oreilles et la queue, par tir de dés consécutifs.

La sortie d'un 6 donne droit de placer au choix une patte, un oeil ou une oreille. La sortie d'un double 6 donne droit de placer la queue.

Votre adversaire est MO5/TO7.

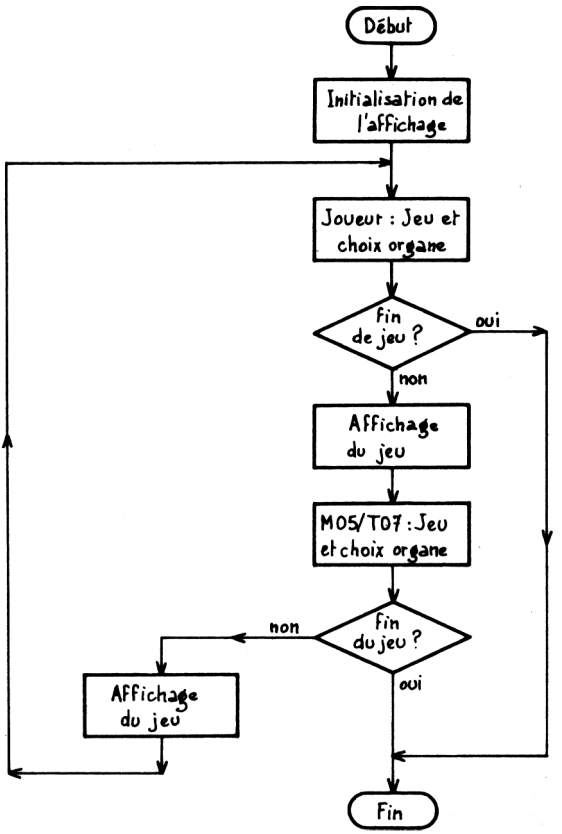
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à troix niveaux définie en annexe.

Le module de traitement consiste en :

- initialisation de l'affichage ;
- jeu utilisateur ;
- test fin de jeu ;
- jeu ordinateur ;
- test fin de jeu.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Présentation et initialisation.
lignes 2000 à 3060 : Jeu.
lignes 4000 à 4110 : Fin du jeu.

Niveau 3 - **lignes 1100 à 1550** : Présentation du jeu.
lignes 1700 à 1830 : Initialisation des variables.
lignes 3100 à 3160 : Tirage aléatoire des dés.

IDENTIFICATION DES VARIABLES

A$: attente d'un caractère pour passer à la suite.

CH : choix d'un organe.

D1, D2 : dés.

G : nombre d'organe(s) placé(s) par le joueur.

I, J : index de boucles.

IRO : indicateur de jeu MO5/TO7.

JOUEUR : indicateur de jeu joueur.

R1, SUIV : choix MO5/TO7.

SI, SO : score joueur, MO5/TO7.

T1, T2 : tableau joueur, MO5/TO7.

T1$: tableau des organes du cochon.

```
10 :   REM Le cochon qui rit
20 :
30 :     GOSUB 1000 'Presentation et Initialisation
40 :     GOSUB 2000 'Jeu
50 :     GOSUB 4000 'Fin du Jeu
60 :
70 :   END
80 :   REM ****************************************
1000 : REM Presentation et Initialisation
1010 :
1020 :   GOSUB 1100 'Presentation
1030 :   GOSUB 1700 'Initialisation
1040 :
1050 : RETURN
1060 : REM ****************************************
1100 : REM Presentation du jeu
1110 :
1120 :   SCREEN 6,0,0 : LOCATE 1,1,0
1130 :
1140 :   REM Titre
1150 :
1160 :     ATTRB 1,1 : COLOR 1 : CLS
1170 :     PRINT:PRINT" LE COCHON QUI RIT"
1180 :
```

```
1190 :    REM Objet
1200 :
1210 :      ATTRB 0,0 : PRINT:PRINT
1220 :      COLOR 4
1230 :      PRINT"OBJET : ";
1240 :      COLOR 6
1250 :      PRINT"Jeu de hasard."
1260 :
1270 :    REM Job
1280 :
1290 :      PRINT:PRINT
1300 :      COLOR 5
1310 :      PRINT"JOB    : ";
1320 :      COLOR 6
1330 :      PRINT"Le Jeu se Joue a 2."
1340 :      PRINT"          L'un des Joueurs est MO5."
1350 :      PRINT"          Le but du Jeu est de construi
re"
1360 :      PRINT"          un cochon auQuel il manQue le
s"
1370 :      PRINT"          yeux,les Pattes,les oreilles
et"
1380 :      PRINT"          la Queue.Le Joueur disPose de
 2"
1390 :      PRINT"          des.Un 6 donne droit a un oei
l,"
1400 :      PRINT"          une Patte,une oreille.Un doub
le"
1410 :      PRINT"          6 donne droit a la Queue."
1420 :      PRINT"          Construirez-vous Plus vite"
1430 :      PRINT"          votre cochon Que MO5 le sien
?"
1440 :
1450 :    REM Debut
1460 :
1470 :      PRINT:PRINT
1480 :      COLOR 3
1490 :      PRINT"DEBUT : ";
1500 :      COLOR 6
1510 :      PRINT"Pour commencer, aPPuyez"
1520 :      PRINT"          sur une touche QuelconQue."
1525 :      A=RND:A$=INKEY$:IF A$="" THEN 1525 'Attente
 de l'aPPui
1530 :
1540 :  RETURN
1550 :  REM ****************************************
1700 :  REM Initialisation des variables du Jeu
1710 :
1720 :    REM OrQanes du cochon
1730 :
```

```
1735 :    MA(1)=4:MA(2)=2:MA(3)=2:MA(4)=1
1740 :    FOR I=1 TO 4
1750 :      READ B$ : T1$(I)=B$
1760 :      T1(I)=0
1770 :      T2(I)=0
1780 :    NEXT I
1790 :
1800 :    DATA Patte (s),Oreille (s),Oeil (Yeux),Queue
1810 :
1820 : RETURN
1830 : REM ****************************************
2000 : REM Jeu
2010 :
2015 :    REM Initialisation de l'affiche
2016 :
2020 :      CLS
2030 :
2040 :      LOCATE 18,2:PRINTCHR$(27)"ADES"
2045 :      COLOR 6
2050 :      LOCATE 5,2:PRINT"0 - 0":LOCATE 31,2:PRINT"0
 - 0"
2060 :      LOCATE 6,0:PRINT"VOUS":LOCATE 32,0:PRINT"MO
5"
2070 :      FOR I=1 TO 7
2080 :        PSET(19,I*2+2)"I"
2090 :      NEXT I
2100 :      LOCATE 3,5:PRINT"Votre cochon":LOCATE 5,6:P
RINT"Possede"
2110 :      LOCATE 23,5:PRINT"Le cochon de MO5":LOCATE
28,6:PRINT"Possede"
2120 :
2130 :      FOR I=1 TO 4
2140 :        LOCATE 5,I*2+8:PRINT T1$(I)
2145 :        LOCATE 25,I*2+8:PRINT T1$(I)
2150 :        LOCATE 2,I*2+8:PRINT T1(I)
2155 :        LOCATE 22,I*2+8:PRINT T2(I)
2160 :      NEXT I
2200 :
2210 :    REM Jeu du joueur
2220 :
2230 :      LOCATE 5,20:COLOR 1:PRINT"Appuyez sur ENTRE
E pour jouer."
2240 :      A$=INPUT$(1)
2245 :      LOCATE 1,20:PRINT CHR$(24)
2250 :      GOSUB 3000 'Tirage aleatoire
2260 :      IF D1<>6 AND D2<>6 THEN 2600 'Coup pour rie
n
2270 :      COLOR 6 : LOCATE 0,18
2280 :      PRINT"Selectionnez un des organes manquant.
"
```

```
2290 :      COLOR 4 : PRINT
2300 :      PRINT"1=Patte, 2=Oreille, 3=Oeil, 4=Queue"
2310 :      PRINT : COLOR 6 : INPUT"Choix";CH
2320 :      IF D1+D2<>12 AND CH=4 THEN PRINTCHR$(11);CH
R$(11);:GOTO 2310
2330 :      IF T1(CH)=MA(CH) THEN 2510 ELSE T1(CH)=T1(C
H)+1
2340 :
2350 :      REM Verification fin de jeu
2360 :
2370 :        FOR I=1 TO 4
2380 :          IF T1(I)=MA(I) THEN G=G+1
2390 :        NEXT I
2400 :        IF G=4 THEN JOUEUR=1 'Partie 9a9nee par l
e joueur
2410 :        G=0 'Reinitialisatioin du compteur
2420 :
2430 :      REM Afficha9e des or9anes
2440 :
2450 :        FOR I=1 TO 4
2460 :          LOCATE 2,I*2+8:PRINT T1(I)
2470 :        NEXT I
2480 :
2490 :      REM Effacement des Informations affichees
2500 :
2510 :        LOCATE 0,18
2520 :        FOR I=1 TO 5
2530 :          PRINT CHR$(24)
2540 :        NEXT I
2550 :
2600 : REM Jeu de MO5
2610 :
2620 :    GOSUB 3100 'Tira9e aleatoire de MO5
2630 :    IF D1<>6 AND D2<>6 THEN 2900 'Coup pour rien
2640 :
2670 :    REM Choix de l'or9ane
2680 :
2690 :      RI=RI+1
2700 :      IF RI<5 THEN SUIV=1
2710 :      IF RI=5 OR RI=6 THEN SUIV=2
2720 :      IF RI=7 OR RI=8 THEN SUIV=3
2730 :      IF RI>=9 THEN SUIV=4
2740 :
2750 :      IF D1+D2=12 AND T2(4)=0 THEN T2(4)=1:RI=RI-
1
2755 :      IF D1+D2<>12 AND SUIV<>4 THEN T2(SUIV)=T2(S
UIV)+1
2760 :
2770 :    REM Affiche9e de l'etat du cochon de MO5
2780 :
```

```
2790 :      FOR I=1 TO 4
2800 :        LOCATE 22,I*2+8:PRINT T2(I)
2810 :      NEXT I
2820 :
2830 :    REM Test si MO5 a 9a9ne
2840 :
2850 :      FOR I=1 TO 4
2860 :        IF T2(I)=MA(I) THEN G=G+1
2870 :      NEXT I
2880 :      IF G=4 THEN IRO=1
2890 :      G=0 'Reinitialisation du compteur
2900 :
2910 :    REM Boucle de jeu
2915 :
2920 :      IF IRO=1 OR JOUEUR=1 THEN RETURN ELSE 2200
2930 : REM ****************************************
3000 : REM Tira9e aleatoire des 2 des
3010 :
3015 :    COLOR 6
3020 :    D1=INT(RND*6)+1 : D2=INT(RND*6)+1
3030 :    LOCATE 4,2,0 : PRINT D1
3035 :    LOCATE 8,2,0 : PRINT D2
3040 :
3050 : RETURN
3060 : REM ****************************************
3100 : REM Tira9e aleatoire MO5
3110 :
3115 :    COLOR 6
3120 :    D1=INT(RND*6)+1 : D2=INT(RND*6)+1
3130 :    LOCATE 30,2,0 : PRINT D1
3135 :    LOCATE 34,2,0 : PRINT D2
3140 :
3150 : RETURN
3160 : REM ****************************************
4000 : REM Fin du jeu
4010 :
4020 :    CLS
4030 :    IF JOUEUR=1 AND IRO=0 THEN PRINT"Pas mal Joue
 !"
4040 :    IF IRO=1 AND JOUEUR=0 THEN PRINT"MO5 est deci
dement tres fort !!"
4045 :    IF IRO=1 AND JOUEUR=1 THEN PRINT"E9alite !"
4050 :    PRINT:PRINT:PRINT
4060 :    PRINT"Score ...":PRINT
4070 :    SJ=T1(1)+T1(2)+T1(3)+T1(4)
4080 :    SO=T2(1)+T2(2)+T2(3)+T2(4)
4090 :    PRINT"Joueur :"SJ"MO5 :"SO
4100 :
4110 : RETURN
```

JEUX DE RÉFLEXION

Jeu des allumettes

PRESENTATION

Ce jeu se joue à deux ou plus de deux joueurs.

C'est un classique parmi le jeux de réflexion.

N allumettes (où $N \geqslant 8$) sont placées sur une table. Chaque joueur retire, à tour de rôle, une, deux ou trois allumettes. Celui qui retire la ou les dernière(s) allumette(s) a gagné.

Ce programme oppose deux joueurs : vous et MO5/TO7.

Bonne chance...

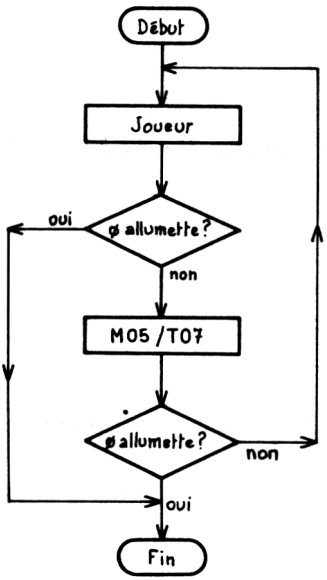
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- jeu utilisateur ;
- test fin de partie ;
- jeu ordinateur ;
- test fin de partie.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1050** : Présentation et initialisation.
lignes 2000 à 2160 : Jeu.
lignes 3000 à 3030 : Fin du jeu.

Niveau 3 - **lignes 1100 à 1480** : Présentation du jeu.
lignes 1600 à 1700 : Initialisation des variables.

IDENTIFICATION DES VARIABLES

A$: attente d'un caractère pour passer à la suite.

GAGN : indique le vainqueur.

N : nombre d'allumettes restantes.

TIR : nombre d'allumettes retirées par MO5/TO7.

USER : nombre d'allumettes retirées par le joueur.

Remarque

Si, au bout d'un certain temps, vous arrivez à battre MO5/TO7 à coup sûr, essayez de modifier sa tactique de jeu (**ligne 2090**).

La tactique de jeu employée par le programme est la suivante :

- s'il reste plus de 7 allumettes, MO5/TO7 en retire un nombre quelconque entre 1 et 3 (utilisation de la fonction RND).
- s'il en reste 7 ou moins :
  - il en reste 7 : MO5/TO7 en prend 3 et gagne en 2 coups ;
  - il en reste 6 : MO5/TO7 en prend 2 et gagne en 2 coups ;
  - il en reste 5 : MO5/TO7 en prend 1 et gagne en 2 coups ;
  - il en reste 4 : MO5/TO7 en prend 1 et vous gagnez ;
  - il en reste 3 : MO5/TO7 gagne en 1 coup.

```
10 :   REM Jeu des allumettes
20 :
30 :     GOSUB 1000 'Presentation et Initialisation
40 :     GOSUB 2000 'Jeu
50 :     GOSUB 3000 'Fin du Jeu
60 :
70 :   END
80 :   REM ****************************************
1000 : REM Presentation et Initialisation
1010 :
1020 :   GOSUB 1100 'Presentation
1030 :   GOSUB 1600 'Initialisation
1040 :
1050 : RETURN
1060 : REM ****************************************
1100 : REM Presentation
1110 :
1120 :   SCREEN 6,0,0 : LOCATE 1,1,0
1130 :
1140 :   REM Titre
1150 :
1160 :     ATTRB 1,1 : COLOR 1,0 : CLS
1170 :     PRINT : PRINT" JEU DES ALLUMETTES"
1180 :
1190 :   REM Objet
1200 :
1210 :     ATTRB 0,0 : PRINT:PRINT:PRINT
1220 :     COLOR 4
1230 :     PRINT"OBJET : ";
1240 :     COLOR 6
```

```
1250 :      PRINT"Jeu de raPidite."
1260 :
1270 :    REM Job
1280 :
1290 :      PRINT:PRINT:PRINT
1300 :      COLOR 5
1310 :      PRINT"JOB    : ";
1320 :      COLOR 6
1330 :      PRINT"N allumettes sont disPosees"
1340 :      PRINT"         sur l'ecran devant 2 Joueurs:
"
1350 :      PRINT"         vous et moi. Chacun des deux"
1360 :      PRINT"         Joueurs va retirer a tour de"
1370 :      PRINT"         role 1, 2 ou 3 allumettes."
1380 :      PRINT"         Celui qui Preleve la ou les"
1390 :      PRINT"         derniere(s) allumette(s)"
1400 :      PRINT"         a 9a9ne."
1410 :
1420 :    REM Debut
1430 :
1440 :      PRINT:PRINT:PRINT
1445 :      COLOR 3
1450 :      PRINT"DEBUT : ";
1455 :      COLOR 6
1460 :      PRINT"Pour commencer, aPPuyez"
1465 :      PRINT"         sur une touche 9uelcon9ue."
1470 :      A=RND:B$=INKEY$:IF B$="" THEN 1470 'Attente
 de l'aPPui
1475 :
1476 : RETURN
1480 : REM ****************************************
1600 : REM Initialisation du Jeu
1610 :
1620 :    CLS 'Effacement d'ecran
1630 :    LOCATE 0,5
1640 :    INPUT"Nombre d'allumettes ";N
1645 :
1650 :    FOR I=1 TO 7
1660 :      READ T(I)
1670 :    NEXT I
1680 :    DATA 1,2,3,1,1,2,3
1685 :
1690 : RETURN
1700 : REM ****************************************
2000 : REM Jeu
2010 :
2020 :    PRINT
2030 :    INPUT"Combien en Prenez-vous ";USER
```

```
2040 :    IF USER<1 OR USER>3 THEN PRINT CHR$(11):GOTO
 2030
2050 :
2060 :    N=N-USER
2070 :    IF N=0 THEN GAGN=1:RETURN
2080 :    PRINT"Il en reste "N
2090 :    IF N>7 THEN TIR=INT(RND*3)+1 ELSE TIR=T(N)
2100 :    PRINT"J'en Prends ";TIR
2110 :    N=N-TIR
2120 :    PRINT"Il en reste "N
2130 :    IF N=0 THEN GAGN=0:RETURN
2140 :    GOTO 2000 'Le jeu continue
2150 :    PRINT"Prenez 1, 2 ou 3 allumettes !!":GOTO 20
00
2155 :
2160 : REM ****************************************
3000 : REM Fin de la Partie
3010 :
3020 :    IF GAGN=0 THEN PRINT"J'ai 9a9ne."ELSE PRINT"V
ous avez 9a9ne."
3025 :
3030 : RETURN
```

Mémoire à court terme

PRESENTATION

Les deux programmes qui suivent (Mémoire à court terme et Simon), testent les capacités de votre mémoire à court terme.

Le premier teste uniquement votre mémoire visuelle, le second teste l'association mémoire auditive - mémoire visuelle.

Peut-être vous demandez-vous ce qu'est la mémoire à court terme ? C'est ce qui nous permet de retenir une suite de concepts de manière immédiate.

MO5/TO7 envoie n caractères à l'écran pendant une seconde, puis les efface. Vous devez restituer les n caractères affichés, dans le bon ordre. Une bonne réponse, et le nombre de caractères augmente d'un au prochain coup. Une mauvaise réponse, et le nombre de caractères diminue d'un au prochain coup.

Au bout d'un certain temps, vous n'arriverez plus à restituer la suite des n caractères. Alors, vous aurez atteint la limite supérieure de votre mémoire à court terme.

ORGANIGRAMME DU JEU

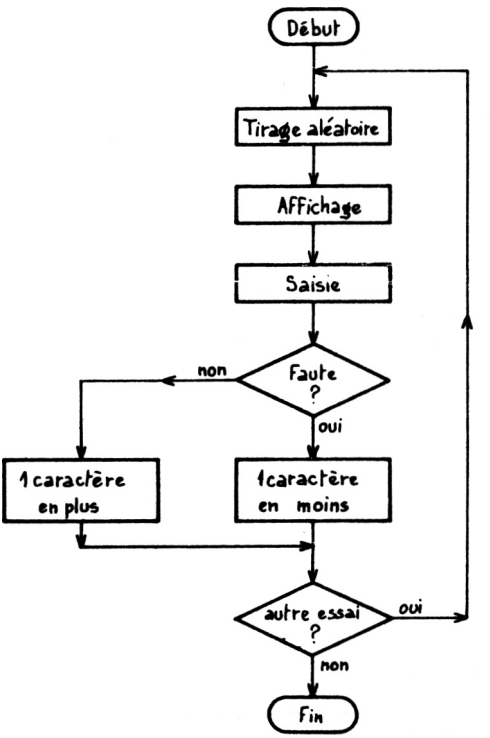
Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- tirage aléatoire des caractères ;
- affichage des caractères ;
- saisie des caractères tapés par l'utilisateur ;

- test si faute ;
- test si autre essai.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 90** : Module de commande.

Niveau 2 - **lignes 1000 à 1380** : Initialisation.
lignes 2000 à 2250 : Jeu.
lignes 3000 à 3120 : Fin de partie.

Niveau 3 - **lignes 2260 à 2350** : Tirage aléatoire.

IDENTIFICATION DES VARIABLES

A : code ASCII du tirage aléatoire.

A$: attente d'un caractère pour passer à la suite.

B : un des caractères de la réponse.

DI : nombre de caractères à afficher.

R$: indicateur de fin de partie.

T : tableau de jeu.

```
10 :   REM Memoire a court terme
20 :
30 :     GOSUB 1000 'Initialisation
40 :     GOSUB 2000 'Jeu
50 :     GOSUB 3000 'Fin de Partie
60 :     IF BIS=1 THEN 40 'Nouvelle Partie
70 :
80 :   END
90 :   REM ****************************************
100 :
1000 : REM Initialisation
1010 :
1020 :   SCREEN 6,0,0 : LOCATE 1,1,0
1030 :
1040 :   REM Titre
1050 :
1060 :     ATTRB 0,1 : COLOR 1,0 : CLS
1070 :     PRINT : PRINT SPC(10)"MEMOIRE A COURT TERME
"
1080 :
1090 :   REM Objet
1100 :
1110 :     ATTRB 0,0 : PRINT:PRINT:PRINT
1120 :     COLOR 4
1130 :     PRINT"OBJET : ";
1140 :     COLOR 6
1150 :     PRINT"Jeu de memoire."
1160 :
1170 :   REM Job
1180 :
1190 :     PRINT:PRINT:PRINT
```

```
1200 :      COLOR 5
1210 :      PRINT"JOB    : ";
1220 :      COLOR 6
1230 :      PRINT"MO5 va afficher une suite"
1240 :      PRINT"         de lettres pendant 3 secondes
."
1250 :      PRINT"         Vous devrez la restituer just
e."
1260 :      PRINT"         Une bonne reponse augmente de
 1"
1270 :      PRINT"         le nombre de lettres, et une"
1280 :      PRINT"         mauvaise reponse le diminue d
'1."
1290 :
1300 :    REM Debut
1310 :
1320 :      PRINT:PRINT:PRINT
1325 :      COLOR 3
1330 :      PRINT"DEBUT  : ";
1335 :      COLOR 6
1340 :      PRINT"Pour commencer, appuyez sur"
1345 :      PRINT"         une touche quelconque."
1350 :
1360 :      A=RND:A$=INKEY$:IF A$="" THEN 1360 'Attente
 de l'appui
1370 :      GA=1  : DI=0 'Affichage d'une lettre au depa
rt
1375 :
1380 :  RETURN
1390 :  REM ****************************************
2000 :  REM Jeu
2010 :
2020 :    IF GA=1 THEN DI=DI+1 ELSE DI=DI-1
2021 :    IF DI<1 THEN DI=1 'Une lettre minimim
2025 :    CLS : LOCATE 10,12,0 : COLOR 6
2030 :    FOR I=1 TO DI
2040 :      GOSUB 2300 'Tirage aleatoire
2050 :      T(I)=A 'Memorisation
2060 :      PRINT CHR$(A);
2070 :    NEXT I
2080 :
2090 :    FOR I=1 TO 1000:NEXT I 'Attente
2100 :
2110 :    CLS
2120 :    LOCATE 0,12
2130 :
2140 :    PRINT CHR$(27)"ATapez la suite de lettre(s)"
2150 :    PRINT CHR$(27)"Apparue sur l'ecran."
```

```
2160 :    PRINT : INPUT A$
2170 :
2175 :    IF DI<>LEN(A$) THEN EN=1:GA=1:RETURN
2180 :    FOR I=1 TO DI
2190 :      B$=MID$(A$,I,1)
2200 :      IF ASC(B$)<>T(I) THEN E=E+1
2210 :    NEXT I
2220 :
2230 :    IF E=0 THEN GA=1 ELSE GA=0
2240 :
2250 : RETURN
2260 : REM ****************************************
2300 : REM Tira9e aleatoire d'une lettre
2310 :
2320 :    A=INT(RND*26)+65
2330 :
2340 : RETURN
2350 : REM ****************************************
3000 : REM Fin de la Partie
3010 :
3020 :    CLS : COLOR 3
3030 :    IF GA=1 AND EN=0 THEN PRINT"Bien Joue !"
3040 :    IF GA=0 AND EN=0 THEN PRINT"Pas bien brillant
 :"E"Erreur(s) !":E=0
3050 :    IF EN=1 THEN EN=0:PRINT"J'avais dit"DI" lettr
e(s) !!!":DI=DI-1
3060 :    PRINT:PRINT:PRINT
3070 :    INPUT"Un autre essai (O/N)";R$
3080 :    IF R$="O" THEN BIS=1 : GOTO 3110 ELSE BIS=0
3090 :    PRINT:PRINT"Fin du jeu au score de"DI"lettre(
s).":COLOR 6
3100 :
3110 : RETURN
3120 : REM ****************************************
```

Simon

PRESENTATION

La mémoire, chez l'être humain, est d'autant plus facilement sollicitée qu'elle a été bien enregistrée.

Elle est d'autant mieux enregistrée qu'elle utilise de sens. Ainsi, le programme "Mémoire à court terme" n'utilisait que la vue pour vous demander de retenir une suite de caractères. Le programme "Simon" utilise vue et ouïe pour retenir une suite de symboles. Votre score devrait donc être supérieur pour ce second programme.

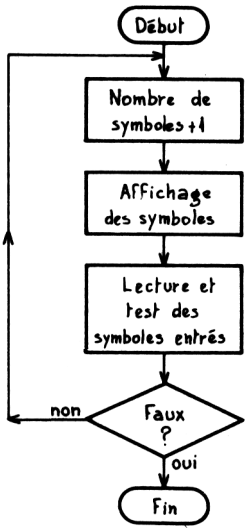
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- incrémentation du nombre de symboles à afficher ;
- affichage de ces symboles ;
- lecture et test des symboles entrés par l'utilisateur.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Présentation et initialisation.
lignes 2000 à 2270 : Jeu.
lignes 3000 à 3090 : Fin du jeu.

Niveau 3 - **lignes 1100 à 1340** : Présentation du jeu.
lignes 1500 à 1900 : Initialisation des variables.
lignes 2300 à 2390 : Affichage case bien sélectionnée.
lignes 2400 à 2490 : Affichage case mal sélectionnée.
lignes 2500 à 2610 : Test si réponse proposée correcte.

IDENTIFICATION DES VARIABLES

A : variable intermédiaire de calcul.

A$: attente d'un caractère pour passer à la suite.

BIS : indicateur de fin de partie.

CO : tableau couleur.

I	: index de boucles.
LO	: tableau coordonnées rectangles.
N	: nombre de symboles à afficher.
NO$	: tableau des notes associées aux rectangles.
OK	: variable de test de réponse joueur.
P	: numéro du rectangle sélectionné par le joueur.
X, Y	: coordonnées du stylo sur l'écran.
X1, Y1, X2, Y2	: diagonale d'un rectangle à afficher.

```
10 :    REM SIMON
20 :
30 :      GOSUB 1000 'Presentation et Initialisation
40 :      GOSUB 2000 'SIMON
50 :      GOSUB 3000 'Fin de la Partie
60 :      IF BIS=-1 THEN 40 'Nouvelle Partie
70 :
75 :    END
80 :    REM ****************************************
1000 :  REM Presentation et Initialisation
1010 :
1020 :    GOSUB 1100 'Presentation du Jeu
1030 :    GOSUB 1500 'Initialisation des variables
1040 :
1050 :  RETURN
1060 :  REM ****************************************
1100 :  REM Presentation du Jeu
1110 :
1115 :    SCREEN 6,0,0 : LOCATE 1,1,0
1116 :
1120 :    REM Titre
1125 :
1130 :      ATTRB 1,1 : COLOR 1 : CLS
1140 :      PRINT:PRINT SPC(7)"SIMON"
1150 :
1160 :    REM Objet
1165 :
1170 :      ATTRB 0,0 : COLOR 4
1175 :      PRINT:PRINT:PRINT:PRINT
1180 :      PRINT"OBJET : ";
1190 :      COLOR 6 : PRINT"Jeu de memoire."
```

```
1200 :
1210 :    REM Job
1215 :
1220 :       PRINT:PRINT:PRINT:PRINT
1225 :       COLOR 5:PRINT"JOB    : ";
1230 :       COLOR 6:PRINT"MO5 va afficher une suite"
1240 :       PRINT"          de symboles lumineux. Grace a
u "
1245 :       PRINT"          stylo optique, vous devrez le
s"
1250 :       PRINT"          restituer dans le bon ordre."
1260 :
1270 :    REM Debut
1275 :
1276 :       PRINT:PRINT:PRINT:PRINT
1280 :       COLOR 3:PRINT"DEBUT  : ";
1290 :       COLOR 6:PRINT"Pour commencer, appuyez"
1300 :       PRINT"          sur une touche quelconque."
1310 :       A=RND:A$=INKEY$:IF A$="" THEN 1310 'Attente
1320 :
1330 : RETURN
1340 : REM ****************************************
1500 : REM Initialisation des variables
1510 :
1515 :    DIM LO(16),T(100)
1516 :
1520 :    REM Couleur des rectangles
1525 :
1530 :       FOR I=1 TO 4
1540 :         READ CO(I)
1550 :       NEXT I
1560 :
1570 :    REM Coordonnees des rectangles
1575 :
1580 :       FOR I=1 TO 16
1590 :         READ LO(I)
1600 :       NEXT I
1610 :
1620 :    REM Notes
1625 :
1630 :       FOR I=1 TO 4
1640 :         READ NO$(I)
1650 :       NEXT I
1660 :
1670 :    REM DATA Couleur rectangles, Coordonnees et N
otes
1675 :
1680 :       DATA 5,4,3,2
1690 :       DATA 125,15,175,50,240,50,290,85
```

```
1695 :      DATA 125,85,175,120,10,50,60,85
1700 :      DATA DO,RE,MI,FA
1710 :
1720 :    REM Initiailisation de l'affiche9e
1730 :
1740 :      CLS 'Effacement d'ecran
1745 :      ATTRB 1,1:COLOR 1:LOCATE 14,18:PRINT"SIMON"
:ATTRB 0,0
1750 :
1760 :      REM Trace et remPlissa9e des rectan9les
1770 :      FOR I=1 TO 4
1780 :        A=(I-1)*4+1
1790 :        COLOR CO(I)
1800 :        BOX(LO(A),LO(A+1))-(LO(A+2),LO(A+3))
1810 :        BOXF(LO(A),LO(A+1))-(LO(A+2),LO(A+3))
1820 :      NEXT I
1830 :
1840 :    REM Messa9e au Joueur
1845 :
1850 :      COLOR 3 : LOCATE 0,23,0
1860 :      PRINT" APPuyez sur une touche Pour commence
r"
1870 :      A$=INPUT$(1)
1880 :
1890 : RETURN
1900 : REM ****************************************
2000 : REM Jeu
2010 :
2020 :    REM Effacement du messa9e
2030 :      LOCATE 0,23:PRINT CHR$(24)
2040 :
2050 :    REM Nombre de tira9es + 1
2060 :      N=N+1 : A=INT(RND*4)+1 : T(N)=A
2070 :
2080 :    REM Affiche9e de la suite a restituer
2090 :    FOR I=1 TO N
2100 :      A=(T(I)-1)*4+1
2110 :      X1=LO(A):Y1=LO(A+1):X2=LO(A+2):Y2=LO(A+3)
2120 :      GOSUB 2300 'Affiche9e de la case selectionn
ee
2130 :    NEXT I
2140 :
2150 :    REM Saisie rePonse et test rePonse O.K.
2160 :
2170 :      FOR I=1 TO N
2180 :        INPUT PEN X,Y
2190 :        GOSUB 2500 'Test rePonse O.K.
2195 :      A=(P-1)*4+1
```

```
2196 :       X1=LO(A):Y1=LO(A+1):X2=LO(A+2):Y2=LO(A+3)
2200 :         IF OK=-1 THEN GOSUB 2300 ELSE GOSUB 2400
'Affiche
2210 :       NEXT I
2220 :
2230 :    REM Boucle de jeu
2240 :      IF OK=-1 THEN 2050
2250 :
2260 :  RETURN
2270 :  REM ****************************************
2300 :  REM Affichage d'une case bien selectionnee
2310 :
2320 :    COLOR 1
2330 :    BOX(X1,Y1)-(X2,Y2)
2335 :    BOXF(X1,Y1)-(X2,Y2)
2340 :    PLAY NO$(T(I))
2350 :    COLOR CO(T(I)) 'Couleur d'origine
2360 :    BOXF(X1,Y1)-(X2,Y2)
2370 :
2380 :  RETURN
2390 :  REM ****************************************
2400 :  REM Affichage d'une case mal selectionnee
2410 :
2420 :    A=(P-1)*4+1
2430 :    X1=LO(A):Y1=LO(A+1):X2=LO(A+2):Y2=LO(A+3)
2440 :    COLOR 1:BOXF(X1,Y1)-(X2,Y2)
2450 :    PLAY"SISISISISI"
2460 :    COLOR CO(P)
2470 :    BOXF(X1,Y1)-(X2,Y2)
2475 :    I=N 'MO5 n'attend Plus de caracteres
2480 :
2485 :  RETURN
2490 :  REM ****************************************
2500 :  REM Test si reponse correcte
2510 :  REM Si oui, OK=-1.Si non,OK=1
2520 :
2530 :    IF Y<51 AND Y>14 AND X<151 AND X>99 THEN P=1
2540 :    IF Y<86 AND Y>49 AND X<291 AND X>239 THEN P=2
2550 :    IF Y<121 AND Y>84 AND X<151 AND X>124 THEN P=
3
2560 :    IF Y<86 AND Y>49 AND X<61 AND X>9 THEN P=4
2570 :
2580 :    IF T(I)=P THEN OK=-1 ELSE OK=1
2590 :
2600 :  RETURN
2610 :  REM ****************************************
3000 :  REM Fin du jeu
3010 :
```

```
3020 :    COLOR 3
3030 :    LOCATE 0,21:PRINT"Perdu au bout de "N" symbol
es."
3040 :    PRINT"Une autre Partie(O/N)"
3050 :    A$=INPUT$(1)
3060 :    IF A$="O" THEN BIS=-1 ELSE BIS=1
3064 :    REM Effacage des informations
3065 :      PRINTCHR$(11);CHR$(11);CHR$(24):PRINT CHR$(
24)
3066 :      N=0 '1 symbole affiche
3067 :      COLOR 6
3070 :
3080 : RETURN
3090 : REM ****************************************
```

Mastermind

PRESENTATION

Le chapitre sur les jeux de réflexion ne saurait être complet sans le très célèbre jeu de MASTERMIND.

Un nombre de quatre chiffres doit être découvert par le joueur en un minimum de coups. MO5/TO7 choisit ce nombre, et indique à chaque essai le nombre de chiffre(s) bien placé(s), et le nombre de chiffre(s) mal placé(s).

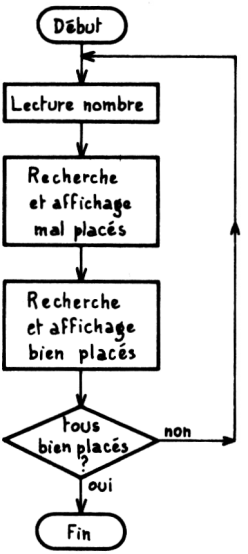
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- lecture du nombre proposé ;
- recherche et affichage chiffre(s) mal placé(s) ;
- recherche et affichage chiffre(s) bien placé(s) ;
- test si nombre trouvé.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Initialisation de la partie.
lignes 2000 à 2200 : Jeu
lignes 3000 à 3060 : Fin du jeu.

Niveau 3 - **lignes 1100 à 1450** : Affichage des règles du jeu.
lignes 1600 à 1710 : Initialisation des variables.

IDENTIFICATION DES VARIABLES

A : indicateur de double dans le nombre à trouver.

A$: attente d'un caractère pour passer à la suite.

BP : indicateur de chiffre(s) bien placé(s).

I, J : index de boucles.

MP : indicateur de chiffre(s) mal placé(s).

NB : nombre de coups joués.

R : tableur joueur : contient le nombre proposé.

T : tableau MO5/TO7 : contient le nombre à trouver.

```
10 :    REM MASTERMIND
20 :
30 :      GOSUB 1000 'Presentation et Initialisation
40 :      GOSUB 2000 'Jeu
50 :      GOSUB 3000 'Fin de la Partie
60 :
70 :    END
80 :    REM ****************************************
1000 :  REM Presentation et Initialisation
1010 :
1020 :    GOSUB 1100 'Presentation
1030 :    GOSUB 1600 'Initiaisation
1040 :
1050 :  RETURN
1060 :  REM ****************************************
1100 :  REM 'Presentation du Jeu
1110 :
1120 :    SCREEN 6,0,0 : LOCATE 1,1,0
1130 :
1140 :    REM Titre
1150 :
1160 :      ATTRB 1,1 : COLOR 1 : CLS
1170 :      PRINT:PRINT"      MASTERMIND"
1180 :
1190 :    REM Objet
1200 :
1210 :      ATTRB 0,0 : PRINT:PRINT:PRINT:PRINT
1220 :      COLOR 4
1230 :      PRINT"OBJET : ";
1240 :      COLOR 6
1250 :      PRINT"Jeu de reflexion."
1260 :
1270 :    REM Job
1280 :
1290 :      COLOR 5 : PRINT:PRINT:PRINT
1300 :      PRINT"JOB    : ";
1310 :      COLOR 6
1320 :      PRINT"MO5 choisit un nombre"
1325 :      PRINT"         a 4 chiffres tous differents.
"
```

→

```
1330 :      PRINT"         Le but du jeu est de trouver
ce"
1335 :      PRINT"         nombre en un minimum de coups
."
1340 :
1345 :    REM Debut
1350 :
1360 :      COLOR 3 : PRINT:PRINT:PRINT
1370 :      PRINT"DEBUT : ";
1380 :      COLOR 6
1390 :      PRINT"Pour commencer, appuyez sur"
1400 :      PRINT"         une touche quelconque."
1410 :      A=RND:A$=INKEY$:IF A$="" THEN 1410 'Attente
 de l'appui
1420 :
1430 :
1440 : RETURN
1450 : REM ****************************************
1600 : REM Initalisation des variables
1610 :
1620 :    FOR I=1 TO 4
1630 :      T(I)=INT(RND*10)
1635 :      IF I=1 THEN GOTO 1680
1640 :      FOR J=1 TO I-1
1650 :        IF T(J)=T(I) THEN A=1
1660 :      NEXT J
1670 :      IF A=1 THEN I=I-1 : A=0
1680 :    NEXT I
1690 :
1700 : RETURN
1710 : REM ****************************************
2000 : REM Jeu
2010 :
2020 :    CLS
2030 :    LOCATE 20,1:PRINT"BP         MP":LOCATE 3,1:PR
INT"Jeu"
2040 :    BP=0:MP=0:NB=NB+1:LI=LI+1
2050 :    LOCATE 0,LI+2
2055 :    INPUT A$ 'Entree du jeu
2060 :    FOR I=1 TO 4
2070 :      R(I)=ASC(MID$(A$,I,1))-48
2080 :    NEXT I
2090 :    FOR I=1 TO 4
2100 :      IF T(I)=R(I) THEN BP=BP+1 ELSE GOSUB 2170
2110 :    NEXT I
2120 :
2140 :    LOCATE 20,LI+2:PRINT BP 'Affichage bien place
(s)
```

```
2150 :    LOCATE 30,LI+2:PRINT MP 'Affichage mal Place(
s)
2155 :    IF LI/20 = INT(LI/20) THEN LI=0
2160 :    IF BP=4 THEN RETURN ELSE 2040 'Boucle PrincIP
ale
2165 :
2170 :    FOR J=1 TO 4
2180 :      IF R(I)=T(J) THEN MP=MP+1
2190 :    NEXT J
2195 :
2200 : RETURN
2210 : REM ****************************************
3000 : REM Fin du jeu
3010 :
3020 :    CLS
3030 :    PRINT"Le nombre a ete trouve en"NB"coups."
3040 :
3050 :
3060 : RETURN
```

GRAPHIQUES ET MATHÉMATIQUES

Figures de Moivre

PRESENTATION

Les figures obtenues par le glissement d'une droite sur deux courbes planes portent le nom de figures de Moivre.

De très belles figures peuvent être obtenues sur les micro-ordinateurs graphiques, à moindre effort.

Dans la suite, six figures sont proposées :

- MOIVRE 1 : les deux courbes planes sont deux droites d'angle 90°.
- MOIVRE 2 : les deux courbes planes sont deux droites d'angle<90°.
- MOIVRE 3 : composition de quatre figures du type 1.
- MOIVRE 4 : autre composition de quatre figures du type 1.
- MOIVRE 5 : composition de deux figures du type 2 pour former un : triangle.
- MOIVRE 6 : application de "MOIVRE 1" pour former le dessin d'un papillon.

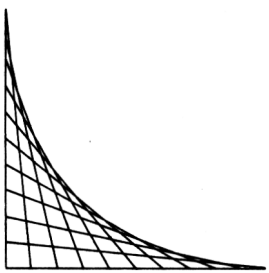
ORGANIGRAMME DES PROGRAMMES

Chacun des six programmes possède la structure linéaire définie en annexe, et consiste en une ou plusieurs boucles décrivant le glissement de la droite.

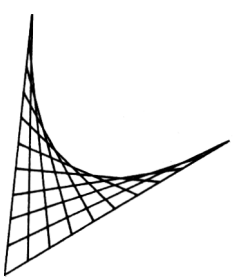
IDENTIFICATION DES VARIABLES

I : index de boucles.

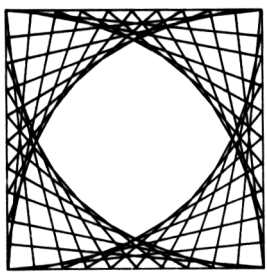
```
10 :  REM MOIVRE 1
20 :
30 :    SCREEN 6,0,0 : CLS
40 :
50 :    FOR I=1 TO 19
60 :      LINE(15,10*I)-(15+10*I,199)
70 :    NEXT I
```



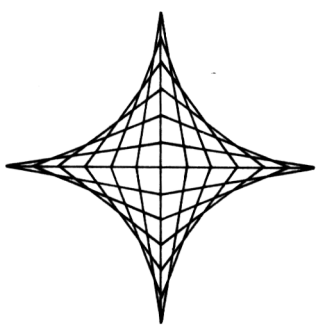
```
10 :  REM MOIVRE 2
20 :
30 :    SCREEN 6,0,0 : CLS
40 :
50 :    FOR I=1 TO 19
60 :      LINE(15,10*I)-(15+10*(I+1),199-9*I)
70 :    NEXT I
```



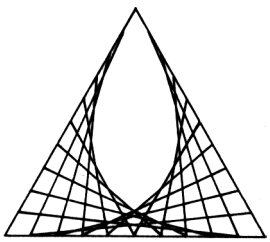
```
10 :  REM MOIVRE 3
20 :
30 :    SCREEN 6,0,0 : CLS
35 :
40 :    FOR I=0 TO 11
50 :      LINE(60,16*I)-(60+16*I,199)
60 :      LINE(60+I*16,0)-(260,16*I)
70 :      LINE(60,199-16*I)-(60+16*I,0)
80 :      LINE(60+16*I,199)-(260,199-16*I)
90 :    NEXT I
100 :   LINE(260,0)-(260,208)
```



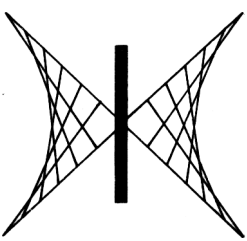
```
10 : REM MOIVRE 4
20 :
30 :   SCREEN 6,0,0 : CLS
40 :
50 :   FOR I=1 TO 11
60 :     LINE(60+I*9,99)-(160,99+9*I)
70 :     LINE(160,99+9*I)-(260-9*I,99)
80 :     LINE(60+9*I,99)-(160,99-9*I)
90 :     LINE(160,99-9*I)-(260-9*I,99)
100 :  NEXT I
```



```
10 : REM MOIVRE 5
20 :
30 :   SCREEN 6,0,0 : CLS
40 :
50 :   FOR I=1 TO 10
60 :     LINE(110+I*10,150)-(165-I*5,30+I*10)
70 :     LINE(210-I*10,150)-(155+I*5,30+I*10)
80 :   NEXT I
90 :   LINE(105,150)-(215,150)
100 :  LINE(215,150)-(160,40)
110 :  LINE(160,40)-(105,150)
```



```
10 : REM MOIVRE 6, PaPillon
20 :
30 :   SCREEN 6,0,0 : CLS
40 :
50 :   FOR I=0 TO 10
60 :     LINE(I*4+120,140-I*4)-(160-I*4,100-I*4)
70 :     LINE(200-I*4,140-I*4)-(160+I*4,100-I*4)
80 :   NEXT I
90 :
100 :  BOX (159,70)-(161,130)
110 :  BOXF(159,70)-(161,130)
```



Formulaire mathématique

PRESENTATION

Cet utilitaire permet de compléter la bibliothèque des fonctions trigonométriques de MO5/TO7.

Cette bibliothèque est actuellement composée de :

COS(a), SIN(a), TAN(a), SQR(a), EXP(a), LOG(a)

L'utilitaire permet de lui rajouter :

$\log_{10}(a)$	: Log en base 10.
SEC(a)	: Secante.
CSC(a)	: Cosecante.
COT(a)	: Cotengente.
ARCSIN(a)	: Arc sinus.
ARCCOS(a)	: Arc cosinus.
ARCSEC(a)	: Arc sécante.
SINH(a)	: sinus hyperbolique.
COSH(a)	: cosinus hyperbolique.
TANH(a)	: tangente hyperbolique.
SECH(a)	: secante hyperbolique.
CSCH(a)	: cosécante hyperbolique.
COTH(a)	: cotangente hyperbolique.
ARCSINH(a)	: arc sinus hyperbolique.
ARCCOSH(a)	: arc cosinus hyperbolique.
ARCTANH(a)	: arc tangente hyperbolique.
ARCSECH(a)	: arc sécante hyperbolique.
ARCCSCH(a)	: arc cosécante hyperbolique.
ARCCOTH(a)	: arc cotangente hyperbolique.

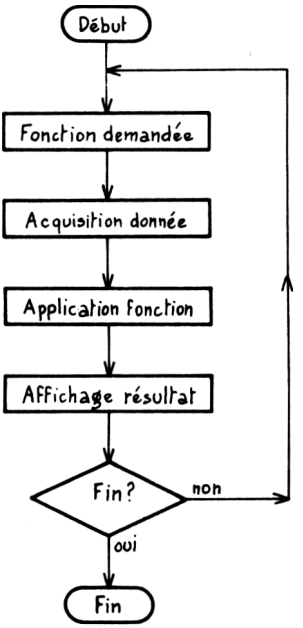
ORGANIGRAMME DU PROGRAMME

Le programme possède la structure hiérarchique à deux niveaux définie en annexe.

Le module de traitement consiste en :

- identification de la fonction demandée ;
- acquisition de la donnée ;
- application de la fonction ;
- affichage du résultat.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1350** : Présentation et initialisation.
lignes 2000 à 3135 : Conversion.
lignes 4000 à 4060 : Fin de conversion.

IDENTIFICATION DES VARIABLES

A$: attente d'un caractère pour passer à la suite.

I : index de boucles.

CH : fonction choisie.

DC : donnée à convertir.

DO : résultat de la fonction appliquée à DC.

AF$: tableau des abréviations des fonctions de la bibliothèque.

R$: variable validant une nouvelle conversion.

```
10 :   REM FORMULAIRE TRIGONOMETRIQUE
20 :
30 :     GOSUB 1000 'Presentation et Initialisation
40 :     GOSUB 2000 'Conversion
50 :     GOSUB 4000 'Fn de conversion
60 :
70 :   END
80 :   REM ****************************************
1000 : REM Presentation et Initialisation
1010 :
1020 :   SCREEN 6,0,0 : LOCATE 1,1,0
1030 :
1040 :   REM Titre
1045 :
1050 :     ATTRB 0,1 : COLOR 1 : CLS
1060 :     PRINT:PRINT SPC(6)"FORMULAIRE TRIGONOMETRIQ
UE"
1070 :
1075 :   REM Objet
1080 :
1085 :     ATTRB 0,0 : COLOR 4 : PRINT:PRINT:PRINT
1090 :     PRINT"OBJET : ";
1095 :     COLOR 6
```

```
1100 :      PRINT"Utilitaire mathematique."
1105 :
1110 :    REM Job
1115 :
1120 :      PRINT:PRINT:PRINT
1125 :      COLOR 5
1130 :      PRINT"JOB    : ";
1135 :      COLOR 6
1140 :      PRINT"Ce programme permet de completer";
1150 :      PRINT"         la bibliotheque des fonctions
"
1160 :      PRINT"         trigonometriques de MO5."
1170 :
1180 :    REM Debut
1190 :
1195 :      PRINT:PRINT:PRINT
1200 :      COLOR 3
1210 :      PRINT"DEBUT : ";
1215 :      COLOR 6
1220 :      PRINT"Pour commencer, appuyez sur"
1225 :      PRINT"         une touche quelconque."
1230 :
1235 :      A$=INPUT$(1) 'Attente d'un caractere
1240 :
1245 :    REM Initialisation
1250 :    REM Lecture des formules des fonctions
1255 :
1256 :      DIM FO$(14)
1260 :      FOR I=1 TO 14
1261 :        READ FO$(I)
1262 :      NEXT I
1270 :    DATA SEC(X)=1/COS(X)
1271 :    DATA CSC(X)=1/SIN(X)
1272 :    DATA COT(X)=1/TAN(X)
1273 :    DATA SINH(X)=(EXP(X)-EXP(-X))/2
1274 :    DATA COSH(X)=(EXP(X)+EXP(-X))/2
1275 :    DATA TANH(X)=(EXP(X)-EXP(-X))/(EXP(X)+EXP(-X)
)
1276 :    DATA SECH(X)=2/(EXP(X)+EXP(-X))
1277 :    DATA CSCH(X)=2/(EXP(X)-EXP(-X))
1278 :    DATA COTH(X)=(EXP(X)+EXP(-X))/(EXP(X)-EXP(-X)
)
1279 :    DATA ARCSINH(X)=LN(X+SQR(X*X+1))
1280 :    DATA ARCCOSH(X)=LN(X+SQR(X*X-1))
1281 :    DATA ARCTANH(X)=LN((1+X)/(1-X))/2
1282 :    DATA ARCSECH(X)=LN((SQR(-X*X+1)+1)/X)
1283 :    DATA ARCCOTH(X)=LN((X+1)/(X-1))/2
```

```
1310 :
1340 : RETURN
1350 : REM ****************************************
2000 : REM Conversion
2010 :
2020 :   REM Menu Principal
2025 :
2030 :     CLS : ATTRB 0,1 : COLOR 1
2035 :     PRINT:PRINT SPC(6)"FORMULAIRE TRIGONOMETRIQ
UE
2040 :
2050 :     ATTRB 0,0 : COLOR 6
2060 :     PRINT:PRINT"Je peux calculer :"
2070 :     PRINT
2080 :     PRINT"(1) Secante"
2081 :     PRINT"(2) Cosecante"
2082 :     PRINT"(3) Cotangente"
2083 :     PRINT"(4) Sinus hyperbolique"
2084 :     PRINT"(5) Cosinus hyperbolique"
2085 :     PRINT"(6) Tangente hyperbolique"
2086 :     PRINT"(7) Secante hyperbolique"
2087 :     PRINT"(8) Cosecante hyperbolique"
2088 :     PRINT"(9) Cotangente hyperbolique"
2089 :     PRINT"(10) Sinus hyperbolique inverse"
2090 :     PRINT"(11) Cosinus hyperbolique inverse"
2091 :     PRINT"(12) Tangente hyperbolique inverse"
2092 :     PRINT"(13) Secante hyperbolique inverse"
2093 :     PRINT"(14) Cotangente hyperbolique inverse
2100 :
2110 :     PRINT:INPUT"Votre choix ";CH
2120 :
2130 :     CLS : PRINT FO$(CH) 'Affichage de la formul
e
2140 :     COLOR 3
2150 :     PRINT:PRINT:INPUT"X =";X
2160 :     ON CH GOSUB 3000,3010,3020,3030,3040,3050,3
060,3070,3080,3090,3100,3110,3120,3130
2170 :
2180 :     PRINT"f(X)=";Y
2190 :     COLOR 6
2200 :     PRINT:PRINT:PRINT
2210 :     INPUT"On continue ";R$
2220 :
2230 :     IF R$="O" THEN 2000
2240 :
2270 : RETURN
2280 : REM ****************************************
```

```
3000 : REM Secante
3001 :
3002 :   Y=1/COS(X)
3003 :
3004 : RETURN
3005 : REM ****************************************
3010 : REM Cosecante
3011 :
3012 :   Y=1/SIN(X)
3013 :
3014 : RETURN
3015 : REM ****************************************
3020 : REM Cotangente
3021 :
3022 :   Y=1/TAN(X)
3023 :
3024 : RETURN
3025 : REM ****************************************
3030 : REM Sinus hyperbolique
3031 :
3032 :   Y=(EXP(X)-EXP(-X))/2
3033 :
3034 : RETURN
3035 : REM ****************************************
3040 : REM Cosinus hyperbolique
3041 :
3042 :   Y=(EXP(X)+EXP(-X))/2
3043 :
3044 : RETURN
3045 : REM ****************************************
3050 : REM Tangente hyperbolique
3051 :
3052 :   Y=(EXP(X)-EXP(-X))/(EXP(X)+EXP(-X))
3053 :
3054 : RETURN
3055 : REM ****************************************
3060 : REM Secante hyperbolique
3061 :
3062 :   Y=2/(EXP(X)+EXP(-X))
3063 :
3064 : RETURN
3065 : REM ****************************************
3070 : REM Cosecante hyperbolique
3071 :
3072 :   Y=2/(EXP(X)-EXP(-X))
3073 :
3074 : RETURN
```

```
3075 : REM ****************************************
3080 : REM Cotangente hyperbolique
3081 :
3082 :   Y=(EXP(X)+EXP(-X))/(EXP(X)-EXP(-X))
3083 :
3084 : RETURN
3085 : REM ****************************************
3090 : REM Arc sinus hyperbolique
3091 :
3092 :   Y=LOG(X+SQR(X*X+1))
3093 :
3094 : RETURN
3095 : REM ****************************************
3100 : REM Arc cosinus hyperbolique
3101 :
3102 :   Y=LOG(X+SQR(X*X-1))
3103 :
3104 : RETURN
3105 : REM ****************************************
3110 : REM Arc tangente hyperbolique
3111 :
3112 :   Y=LOG((1+X)/(1-X))/2
3113 :
3114 : RETURN
3115 : REM ****************************************
3120 : REM Arc secante hyperbolique
3121 :
3122 :   Y=LOG((SQR(-X*X+1)+1)/X)
3123 :
3124 : RETURN
3125 : REM ****************************************
3130 : REM Arc cotangente hyperbolique
3131 :
3132 :   Y=LOG((X+1)/(X-1))/2
3133 :
3134 : RETURN
3135 : REM ****************************************
4000 : REM Fin de conversion
4010 :
4020 :   CLS
4030 :   PRINT"A la prochaine..."
4040 :
4050 : RETURN
4060 : REM ****************************************
```

Le calendrier perpétuel

PRESENTATION

Ce petit utilitaire utilise la formule de GAUSS pour établir la correspondance entre date chiffrée et jour de la semaine.

Après avoir entré la date (jour, mois, année) pour laquelle vous voulez connaître le jour, MO5/TO7 applique ces données à la formule de GAUSS et affiche immédiatement le jour correspondant.

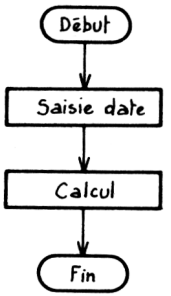
ORGANIGRAMME DU PROGRAMME

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- saisie de la date (jour, mois, année) ;
- application de cette date à la formule de GAUSS.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Présentation et initialisation.
lignes 2000 à 2140 : Traitement.
lignes 3000 à 3060 : Affichage du résultat.

Niveau 3 - **lignes 1100 à 1420** : Présentation.
lignes 1600 à 1670 : Initialisation des variables.

IDENTIFICATION DES VARIABLES

A$: attente d'un caractère pour passer à la suite.

AN$: année considérée.

JOUR : jour considéré.

J$: jour de la semaine.

M, C, A : variables de calcul.

MOIS : mois considéré.

Z : jour calculé.

```
10 :    REM LE CALENDRIER PERPETUEL
20 :
30 :      GOSUB 1000 'Presentation et Initialisation
40 :      GOSUB 2000 'Traitement
50 :      GOSUB 3000 'Fin
60 :
70 :    END
80 :    REM ****************************************
1000 :  REM Presentation et Initialisation
1010 :
1020 :    GOSUB 1100 'Presentation
1030 :    GOSUB 1600 'Initialisation
1040 :
1050 :  RETURN
1060 :  REM ****************************************
1100 :  REM Presentation
1110 :
1120 :    SCREEN 6,0,0 : LOCATE 1,1,0
1130 :
1140 :    REM Titre
1150 :
1160 :      ATTRB 0,1 : COLOR 1 : CLS
1170 :      PRINT:PRINT SPC(9)"LE CALENDRIER PERPETUEL"
1180 :
1190 :    REM Objet
1200 :
1210 :      ATTRB 0,0:PRINT:PRINT:PRINT:PRINT
1220 :      COLOR 4
1230 :      PRINT"OBJET : ";
1240 :      COLOR 6
1250 :      PRINT"Utilitaire."
1260 :
1270 :    REM Job
1280 :
1290 :      PRINT:PRINT:PRINT
1300 :      COLOR 5
1310 :      PRINT"JOB :   ";
1320 :      COLOR 6
1330 :      PRINT"Ce calendrier Permet de"
1335 :      PRINT"         trouver le nom d'un Jour,"
1340 :      PRINT"         connaissant Jour/Mois/Annee."
1345 :      PRINT"         Le calendrier est etabli"
1350 :      PRINT"         a Partir de la formule de GAU
SS."
1355 :
1360 :    REM Debut
1365 :
```

```
1370 :      PRINT:PRINT:PRINT
1375 :      COLOR 3
1380 :      PRINT"DEBUT : ";
1385 :      COLOR 6
1390 :      PRINT"Pour commencer, appuyez sur"
1395 :      PRINT"         une touche quelconque."
1400 :      A$=INPUT$(1)
1405 :
1410 :  RETURN
1420 :  REM ****************************************
1600 :  REM Initialisation des variables
1610 :
1620 :    FOR I=0 TO 6
1630 :      READ J$(I)
1640 :    NEXT I
1645 :
1650 :    DATA Dimanche,Lundi,Mardi,Mercredi,Jeudi,Vend
redi,Samedi
1655 :
1660 :  RETURN
1670 :  REM ****************************************
2000 :  REM Traitement
2010 :
2020 :    CLS : LOCATE 0,5
2030 :    INPUT"Jour ";JOUR
2040 :    PRINT
2050 :    INPUT"Mois ";MOIS
2060 :    PRINT
2070 :    INPUT"Annee ";AN$
2080 :    M=MOIS-2:IF MOIS<=2 THEN M=MOIS+10
2090 :    C=VAL(LEFT$(AN$,2)):A=VAL(RIGHT$(AN$,2))
2095 :    IF MOIS<=2 THEN A=A-1
2100 :    X=INT(2.6*M-0.199)+JOUR+A+INT(A/4)+INT(C/4)-2
*C
2110 :    Z=INT(X)-7*INT(X/7)
2115 :
2120 :  RETURN
2140 :  REM ****************************************
3000 :  REM Affichage du resultat
3010 :
3020 :    PRINT:PRINT:PRINT:COLOR 3
3030 :    PRINT"Le "JOUR"/"MOIS"/"AN$" est un ";
3040 :    COLOR 1:PRINTJ$(Z):COLOR 6
3050 :
3060 :  RETURN
```

Calendrier

PRESENTATION

Ce programme fait appel à la même formule que le précédent. Si vous possédez une imprimante, il vous permettra de reproduire le calendrier d'une année quelconque, mois par mois.

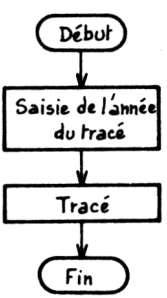
ORGANIGRAMME DU PROGRAMME

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- saisie de l'année du tracé ;
- tracé.

L'organigramme du tracé est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 90** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Présentation et initialisation.
lignes 2000 à 2270 : Tracé du calendrier.
lignes 4000 à 4060 : Validation nouveau tracé.

Niveau 3 - **lignes 1100 à 1340** : Présentation du programme.
lignes 1400 à 1490 : Initialisation des constantes.
lignes 2300 à 2430 : Calcul des premiers jours J(1) et J(2) des mois affichés.
lignes 2500 à 2560 : Affichage ligne 1.
lignes 2600 à 2660 : Affichage ligne 2.
lignes 2700 à 2760 : Affichage ligne 3.
lignes 2800 à 2970 : Affichage ligne 4.
lignes 3000 à 3180 : Affichage ligne>4 et<10.

IDENTIFICATION DES VARIABLES

A, C, IN$, M, SE$, X : variables intermédiaires de calcul.
A$: attente d'un caractère pour passer à la suite.
I : index de boucles.
J : tableau du 1er jour mois impair et pair.
JC : tableau jour courant mois impair et pair.
JOUR : jour appliqué à la formule de GAUSS.
L : ligne courante.
LI$: ligne à imprimer.
MAX : nombre de jours de chaque mois.
MO$: tableau des mois.
MOIS : mois du tracé.
N : numéro de mois impair.
R$: indicateur de nouveau tracé.

```
10 :    REM CALENDRIER
20 :
30 :      GOSUB 1000 'Presentation et Initialisation
40 :      GOSUB 2000 ' Trace du calendrier
50 :      GOSUB 4000 ' Validation eventuelle d'un nouve
au trace
60 :      IF R$="O" THEN 40
70 :
80 :    END
90 :    REM ****************************************
1000 :  REM Presentation et Initialisation
1010 :
1020 :    GOSUB 1100 'Presentation
1030 :    GOSUB 1400 'Initialisation
1040 :
1050 :  RETURN
1060 :  REM ****************************************
1100 :  REM Presentation de l'utilitaire
1110 :
1120 :    SCREEN 6,0,0 : LOCATE 1,1,0
1125 :
1130 :    REM Titre
1135 :
1140 :      ATTRB 1,1 : COLOR 1 : CLS
1145 :      PRINT:PRINT SPC(4)"CALENDRIER"
1150 :
1160 :    REM Objet
1165 :
1170 :      ATTRB 0,0:PRINT:PRINT:PRINT
1175 :      COLOR 4
1180 :      PRINT"OBJET : ";
1185 :      COLOR 6
1190 :      PRINT"Trace de calendrier."
1195 :
1200 :    REM Job
1205 :
1210 :      PRINT:PRINT:PRINT
1215 :      COLOR 5
1220 :      PRINT"JOB :    ";
1225 :      COLOR 6
1230 :      PRINT"Si vous Possedez une imPrimante"
1235 :      PRINT"          80 colonnes, cet utilitaire"
1240 :      PRINT"          tracera Pour vous le calendri
er"
1245 :      PRINT"          d'une annee 9uelcon9ue."
1250 :
1260 :    REM Debut
1265 :
```

```
1270 :      PRINT:PRINT:PRINT
1275 :      COLOR 3
1280 :      PRINT"DEBUT : ";
1285 :      COLOR 6
1290 :      PRINT"Pour commencer, appuyez sur"
1295 :      PRINT"         une touche quelconque."
1300 :
1310 :      A$=INPUT$(1) 'Attente de l'appui
1320 :
1330 : RETURN
1340 : REM ***************************************
1400 : REM Initialisation des constantes du programme
1410 :
1420 :    DIM MO$(12),MAX(12)
1430 :    FOR I=1 TO 12
1440 :      READ MO$(I)
1450 :    NEXT I
1451 :
1452 :    FOR I=1 TO 12
1453 :      READ MAX(I)
1454 :    NEXT I
1455 :
1460 :    DATA Janvier,Fevrier,Mars,Avril,Mai,Juin,Juil
let
1461 :    DATA Aout,Septembre,Octobre,Novembre,Decembre
1462 :
1465 :    DATA 31,28,31,30,31,30,31,31,30,31,30,31
1470 :
1480 : RETURN
1490 : REM ***************************************
2000 : REM Trace du calendrier
2010 :
2020 :    REM Saisie de l'annee du trace
2030 :
2040 :      ATTRB 1,1 : COLOR 1 : CLS
2045 :      PRINT:PRINTSPC(5)"CALENDRIER"
2050 :      ATTRB 0,0 : COLOR 6
2060 :      LOCATE 7,10
2065 :      INPUT"Annee du calendrier ";AN$
2070 :      LOCATE 0,18 : COLOR 3
2075 :      PRINT"  Connectez l'imprimante puis, appuye
z"
2080 :      PRINT"  sur une touche quelconque."
2085 :
2090 :      LOCATE 1,1,0:A$=INPUT$(1) 'Curseur invisibl
e et Attente
2100 :
2120 :    REM Trace
```

```
2125 :
2130 :      MOIS=1 'Mois du trace=Janvier
2140 :      OPEN"O",#2,"LPRT:(80)"
2145 :      PRINT#2,SPC(32)"CALENDRIER "AN$
2150 :      PRINT#2:PRINT#2:PRINT#2
2155 :
2160 :      FOR N=1 TO 6
2170 :        GOSUB 2300 'Calcul des Jours J(1) et J(2)
 et d'annee bisextile
2180 :        FOR L=1 TO 10
2190 :          ON L GOSUB 2500,2600,2700,2800,3000,300
0,3000,3000,3000,2700
2200 :          PRINT #2,LI$
2210 :        NEXT L
2220 :        PRINT#2:PRINT#2 'Espace Inter-Mois
2225 :        MOIS=MOIS+2
2230 :      NEXT N
2240 :      CLOSE #2
2250 :
2260 : RETURN
2270 : REM ************************************
2300 : REM Calcul des Premiers Jours J(1) et J(2) des
mois a afficher
2301 : REM et d'annee bissextile
2310 :
2311 :    AN=VAL(AN$) : SEC=0
2312 :    R$=RIGHT$(AN$,1):L$=LEFT$(AN$,2)
2313 :    IF VAL(R$)=0 THEN SEC=1 ELSE SEC=0 'Annee sec
ulaire
2314 :    BISSEX=0 'Indicateur d'annee bisextile
2315 :    IF SEC=0 AND INT(AN/4)=AN/4 THEN BISSEX=1
2316 :    IF SEC=1 AND INT(VAL(L$)/4)=VAL(L$)/4 THEN BI
SSEX=1
2317 :    IF BISSEX=1 THEN MAX(2)=29 ELSE MAX(2)=28
2318 :
2319 :
2320 :    JOUR=1 'Recherche du 1er Jour
2330 :    FOR I=1 TO 2
2340 :      M=MOIS-2:IF MOIS<=2 THEN M=MOIS+10
2350 :      C=VAL(LEFT$(AN$,2)):A=VAL(RIGHT$(AN$,2))
2360 :      IF MOIS<=2 THEN A=A-1
2370 :      X=INT(2.6*M-0.199)+JOUR+A+INT(A/4)+INT(C/4)
-2*C
2380 :      J(I)=INT(X)-7*INT(X/7)+1
2390 :      MOIS=MOIS+1 'Passa9e au mois suivant
2400 :    NEXT I
2405 :    MOIS=MOIS-2
2410 :
```

```
2420 : RETURN
2430 : REM ****************************************
2500 : REM Affichage 1e ligne
2510 :
2511 :   LI$="       "
2515 :   FOR K=1 TO 2
2520 :     LL=LEN(MO$(MOIS))+5
2525 :     L1=INT((24-LL)/2)
2526 :     FOR I=1 TO L1:LI$=LI$+"-":NEXT I
2527 :     LI$=LI$+MO$(MOIS)+" "+AN$
2528 :     FOR I=1 TO 24-L1-LL:LI$=LI$+"-":NEXT I
2529 :     IF K=1 THEN LI$=LI$+"                    ":MO
IS=MOIS+1
2530 :   NEXT K
2535 :   MOIS=MOIS-1
2540 :
2550 : RETURN
2560 : REM ****************************************
2600 : REM Affichage 2e ligne
2610 :
2620 :   IN$=": D  L  M  M  J  V  S  :"
2630 :   LI$="       "+IN$+"                    "+IN$
2640 :
2650 : RETURN
2660 : REM ****************************************
2700 : REM Calcul 3e ligne
2710 :
2720 :   IN$=":--------------------:"
2730 :   LI$="       "+IN$+"                    "+IN$
2740 :
2750 : RETURN
2760 : REM ****************************************
2800 : REM Calcul 4e ligne
2810 :
2820 :   SE$="   ":LI$="       :"
2830 :   FOR I=1 TO 7
2840 :     IF I<J(1) THEN LI$=LI$+SE$ ELSE LI$=LI$+STR
$(I-J(1)+1)+" "
2860 :   NEXT I
2870 :
2880 :   JC(1)=I-J(1)+1:LI$=LI$+" :                  :
"
2890 :   FOR I=1 TO 7
2900 :     IF I<J(2) THEN LI$=LI$+SE$ ELSE LI$=LI$+STR
$(I-J(2)+1)+" "
2910 :   NEXT I
2920 :
2930 :   LI$=LI$+" :"
```

```
2940 :    JC(2)=I-J(2)+1
2950 :
2960 : RETURN
2970 : REM ****************************************
2980 :
3000 : REM Calcul li9ne > 4
3010 :
3020 :    LI$="       :"
3030 :    FOR I=JC(1) TO JC(1)+6
3035 :       IF I>MAX(MOIS) THEN 3060
3040 :       LI$=LI$+STR$(I)
3050 :       IF I<=9 THEN LI$=LI$+" "
3060 :    NEXT I
3070 :    LL=LEN(LI$):IF LL<28 THEN FOR K=LL TO 27:LI$=
LI$+" ":NEXT K
3080 :
3090 :    JC(1)=I:LI$=LI$+" :                    :"
3100 :    FOR I=JC(2) TO JC(2)+6
3110 :       IF I>MAX(MOIS+1) THEN 3140
3120 :       LI$=LI$+STR$(I)
3130 :       IF I<=9 THEN LI$=LI$+" "
3140 :    NEXT I
3145 :    LL=LEN(LI$):IF LL<70 THEN FOR K=LL TO 69:LI$=
LI$+" ":NEXT K
3150 :    JC(2)=I:LI$=LI$+" :"
3160 :
3170 : RETURN
3180 : REM ****************************************
4000 : REM Validation eventuelle d'un nouveau trace
4010 :
4020 :    CLS : COLOR 6
4030 :    INPUT"Un nouveau trace (O/N) ";R$
4040 :
4050 : RETURN
4060 : REM ****************************************
```

CALENDRIER 1984

Janvier 1984

D	L	M	M	J	V	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Fevrier 1984

D	L	M	M	J	V	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29			

Mars 1984

D	L	M	M	J	V	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Avril 1984

D	L	M	M	J	V	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Mai 1984

D	L	M	M	J	V	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Juin 1984

D	L	M	M	J	V	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Juillet 1984

D	L	M	M	J	V	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Aout 1984

D	L	M	M	J	V	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

```
-----Septembre 1984-----
: D  L  M  M  J  V  S  :
:-------------------:
:                1  :
: 2  3  4  5  6  7  8  :
: 9  10 11 12 13 14 15 :
: 16 17 18 19 20 21 22 :
: 23 24 25 26 27 28 29 :
: 30                   :
:-------------------:
```

```
------Octobre 1984------
: D  L  M  M  J  V  S  :
:-------------------:
:    1  2  3  4  5  6  :
: 7  8  9  10 11 12 13 :
: 14 15 16 17 18 19 20 :
: 21 22 23 24 25 26 27 :
: 28 29 30 31          :
:                      :
:-------------------:
```

```
-----Novembre 1984------
: D  L  M  M  J  V  S  :
:-------------------:
:                1  2  3  :
: 4  5  6  7  8  9  10 :
: 11 12 13 14 15 16 17 :
: 18 19 20 21 22 23 24 :
: 25 26 27 28 29 30    :
:                      :
:-------------------:
```

```
-----Decembre 1984------
: D  L  M  M  J  V  S  :
:-------------------:
:                   1  :
: 2  3  4  5  6  7  8  :
: 9  10 11 12 13 14 15 :
: 16 17 18 19 20 21 22 :
: 23 24 25 26 27 28 29 :
: 30 31                :
:-------------------:
```

CALENDRIER 1985

```
------Janvier 1985------
: D  L  M  M  J  V  S  :
:-------------------:
:       1  2  3  4  5  :
: 6  7  8  9  10 11 12 :
: 13 14 15 16 17 18 19 :
: 20 21 22 23 24 25 26 :
: 27 28 29 30 31       :
:                      :
:-------------------:
```

```
------Fevrier 1985------
: D  L  M  M  J  V  S  :
:-------------------:
:                1  2  :
: 3  4  5  6  7  8  9  :
: 10 11 12 13 14 15 16 :
: 17 18 19 20 21 22 23 :
: 24 25 26 27 28       :
:                      :
:-------------------:
```

```
-------Mars 1985--------
: D  L  M  M  J  V  S  :
:-------------------:
:                1  2  :
: 3  4  5  6  7  8  9  :
: 10 11 12 13 14 15 16 :
: 17 18 19 20 21 22 23 :
: 24 25 26 27 28 29 30 :
: 31                   :
:-------------------:
```

```
-------Avril 1985-------
: D  L  M  M  J  V  S  :
:-------------------:
:    1  2  3  4  5  6  :
: 7  8  9  10 11 12 13 :
: 14 15 16 17 18 19 20 :
: 21 22 23 24 25 26 27 :
: 28 29 30             :
:                      :
:-------------------:
```

Mai 1985

D	L	M	M	J	V	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Juin 1985

D	L	M	M	J	V	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

Juillet 1985

D	L	M	M	J	V	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Aout 1985

D	L	M	M	J	V	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Septembre 1985

D	L	M	M	J	V	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Octobre 1985

D	L	M	M	J	V	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Novembre 1985

D	L	M	M	J	V	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Decembre 1985

D	L	M	M	J	V	S
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Biorythmes

PRESENTATION

Des études de statistique ont permis de modéliser la "forme" relative d'un individu tout au long de sa vie. Cette modélisation met en jeu trois cycles temporels sinusoïdaux :

- un cycle physique de 23 jours ;
- un cycle émotionnel de 28 jours ;
- un cycle intellectuel de 33 jours.

La représentation mensuelle de ces trois cycles porte le nom de "biorythme". Les biorythmes sont particuliers à chaque personne. En effet, leur origine coïncide avec la date de naissance de l'intéressé(e).

L'ordonnée Y de chaque sinusoïde indique :

- une période active si $Y>0$ (période de grande forme) ;
- une période passive si $Y<0$ (période de récupération) ;
- une période critique si $Y=0$ (redoublez d'attention ces jours-là, si possible, restez bien tranquilement chez vous...).

Remarque

Les possesseurs d'imprimante et de MO5 pourront sauvegarder leur biorythme grâce à la commande de hard-copy d'écran SCREENPRINT.

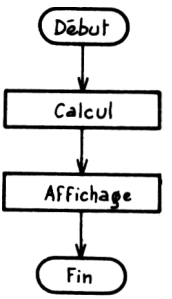
ORGANIGRAMME DU PROGRAMME

Le programme possède la structure hiérarchique à trois niveaux définie en annexe.

Le module de traitement consiste en :

- calcul du nombre de jours écoulés depuis la naissance ;
- affichage des biorythmes.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1060** : Présentation et initialisation.
lignes 2000 à 2180 : Acquisition des données.
lignes 3000 à 3710 : Tracé du biorythme.

Niveau 3 - **lignes 1100 à 1430** : Présentation du programme.
lignes 1600 à 1710 : Initialisation des variables.

IDENTIFICATION DES VARIABLES

A$: attente d'un caractère pour passer à la suite.

A, B, C : ordonnée de chaque cycle.

AB : année du biorythme.

BI : indicateur d'année bisextile.

I : index de boucle.

IN : variable de calcul.

JN : jour de naissance.

MN : mois de naissance.

T : tableau du nombre de jours de chaque mois.

TT : nombre de jours écoulés depuis la naissance.

```
10 :    REM Biorythmes
20 :
30 :      GOSUB 1000 'Presentation et Initialisation
40 :      GOSUB 2000 'Acquisition des donnees
50 :      GOSUB 3000 'Trace des biorythmes
60 :
70 :    END
80 :    REM ****************************************
1000 :  REM Presentation et Initialisation
1010 :
1020 :    GOSUB 1100 'Presentation
1030 :    GOSUB 1600 'Initialisation
1040 :
1050 :  RETURN
1060 :  REM ****************************************
1100 :  REM Presentation
1110 :
1120 :    SCREEN 6,0,0 : LOCATE 1,1,0
1125 :
1130 :    REM Titre
1135 :
1140 :      ATTRB 1,1 : COLOR 1 : CLS
1150 :      PRINT:PRINT"     BIORYTHMES"
1160 :
1165 :    REM Objet
1170 :
1175 :      ATTRB 0,0 : PRINT:PRINT:PRINT
1180 :      COLOR 4
1185 :      PRINT"OBJET : ";
1190 :      COLOR 6
1195 :      PRINT"Trace de biorythmes."
1200 :
1205 :    REM Job
1210 :
1215 :      PRINT:PRINT:PRINT
1220 :      COLOR 5
1225 :      PRINT"Job    : ";
1230 :      COLOR 6
1235 :      PRINT"APres l'entree de votre date"
1240 :      PRINT"         de naissance, des mois et"
1250 :      PRINT"         annee Pour lesquels vous"
1260 :      PRINT"         desirez connaitre les courbes
"
1270 :      PRINT"         biorythmes, MO5 se fera un "
1280 :      PRINT"         Plaisir de tracer les cycles"
1290 :      PRINT"         intellectuel, Phsique et"
1300 :      PRINT"         affectif du mois demande."
1310 :
```

→

```
1320 :    REM Debut
1330 :
1340 :      PRINT:PRINT:PRINT
1350 :      COLOR 3
1360 :      PRINT"DEBUT : ";
1370 :      COLOR 6
1380 :      PRINT"Pour commencer, appuyez sur"
1390 :      PRINT"         une touche quelconque."
1400 :
1410 :      A$=INPUT$(1) 'Attente de l'appui sur une to
uche
1420 :
1430 : RETURN
1440 : REM ****************************************
1600 : REM Initialisation des variables
1610 :
1620 :    REM Duree en jour de chaque mois
1630 :
1640 :      DIM T(12) 'Tableau des mois
1650 :      FOR I=1 TO 12
1660 :        READ T(I)
1670 :      NEXT I
1680 :      DATA 31,28,31,30,31,30,31,31,30,31,30,31
1685 :      PI=3.141592654
1690 :
1700 : RETURN
1710 : REM ****************************************
2000 : REM Acquisition des donnees
2010 :
2020 :    CLS:COLOR 1
2030 :    PRINT SPC(10)"Saisie des donnees"
2040 :    LOCATE 0,6,1
2050 :    COLOR 3:PRINT"Date de naissance:"
2060 :    PRINT
2070 :    INPUT"->Jour de naissance ";JN
2080 :    INPUT"->Mois de naissance ";MN
2090 :    INPUT"->Annee de naissance ";AN
2100 :
2110 :    LOCATE 0,15
2120 :    COLOR 6 : PRINT"Biorythmes :"
2130 :    LOCATE 0,17
2140 :    INPUT"->Mois du trace ";MB
2150 :    INPUT"->Annee du biorythme";AB
2160 :
2170 : RETURN
2180 : REM ****************************************
2385 PRINT A,B,C
2430 :      LOCATE 4,23:PRINT"1":LOCATE 9,23:PRINT"5"
```

```
2435 :      LOCATE 14,23:PRINT"10":LOCATE 19,23:PRINT"1
5"
2440 :      LOCATE 24,23:PRINT"20":LOCATE29,23:PRINT"25
"
2445 :      LOCATE 34,23:PRINT"30"
2450 :      LOCATE 14,23:PRINT"10"
2470 :
3000 : REM Trace des biorythmes
3010 :
3020 :   REM Calcul du nombre de jours ecoules depuis
la naissance
3030 :
3040 :      REM Pendant l'annee de naissance
3050 :        IF (AN/4)=INT(AN/4) THEN IN=1 ELSE IN=0
3060 :        IF IN=1 AND MN=2 THEN FIN=29 ELSE FIN=T(M
N)
3065 :        FOR I=JN TO FIN
3070 :          TT=TT+1
3080 :        NEXT I
3090 :        FOR I=MN+1 TO 12
3100 :          IF IN=1 AND I=2 THEN TT=TT+29 ELSE TT=T
T+T(I)
3110 :        NEXT I
3120 :
3130 :      REM Pendant les annees intermediaires
3140 :        FOR I=2 TO AB-AN
3150 :          B=AN+I-1 'Annee consideree
3160 :          IF B/4 = INT(B/4) THEN TT=TT+1
3170 :          TT=TT+365
3180 :        NEXT I
3190 :
3200 :      REM Pendant l'annee du biorythme
3210 :        IF AB/4 = INT(AB/4) THEN BI=1 ELSE BI=0
3220 :        FOR I=1 TO MB-1
3230 :          IF BI=1 AND I=2 THEN TT=TT+29 ELSE TT=T
T+T(I)
3240 :        NEXT I
3250 :
3260 : REM Affiche9e
3270 :
3280 :    CLS : COLOR 1 : LOCATE 15,1,0
3290 :    PRINT"BIORYTHMES"
3300 :    COLOR 6
3310 :
3320 :    FOR I=1 TO T(MB)
3330 :      A=SIN(2*PI*(TT+I)/23)
3340 :      PSET(I+3,11-INT(9*A))"*"
3350 :      C=SIN(2*PI*(TT+I)/33)
```

```
3360 :      PSET(I+3,11-INT(9*C))"+"
3370 :      B=SIN(2*PI*(TT+I)/28)
3380 :      PSET(I+3,11-INT(9*B))"."
3390 :    NEXT I
3400 :
3410 :    REM Affichage des jours
3420 :
3425 :      COLOR 4
3430 :      LOCATE 4,21:PRINT"1":LOCATE 9,21:PRINT"5"
3440 :      LOCATE 14,21:PRINT"10":LOCATE 19,21:PRINT"1
5"
3450 :      LOCATE 24,21:PRINT"20":LOCATE 29,21:PRINT"2
5"
3460 :      LOCATE 34,21:PRINT"30"
3480 :    REM Axes
3490 :
3495 :      COLOR 6
3500 :      FOR I=1 TO 31
3510 :        PSET(3+I,11)"-"
3520 :      NEXT I
3530 :
3540 :      FOR I=1 TO 20
3550 :        PSET(3,I+1)"I"
3560 :      NEXT I
3570 :
3580 :    REM Nature des courbes
3590 :
3595 :      COLOR 3
3600 :      LOCATE 5,22:PRINT"* = Physique    . = Affec
tif"
3610 :      LOCATE 5,23:PRINT"+ = Intellectuel"
3620 :
3630 :    A$=INPUT$(1) 'Attente d'un caractere
3640 :
3650 :    CLS : LOCATE 15,12,1 : COLOR 6
3660 :    PRINT"A bientot..."
3670 :
3710 : RETURN
```

MO5/TO7
JOUE ET DESSINE

Piano

PRESENTATION

Si vous jouez du piano, ce programme peut vous sembler désuet. Si vous n'en jouez pas, c'est l'occasion de vous y mettre.

Les touches du clavier reprennent la disposition des tons et demi-tons du clavier du piano. Les notes et leur correspondant-touche sont indiquées sur l'écran.

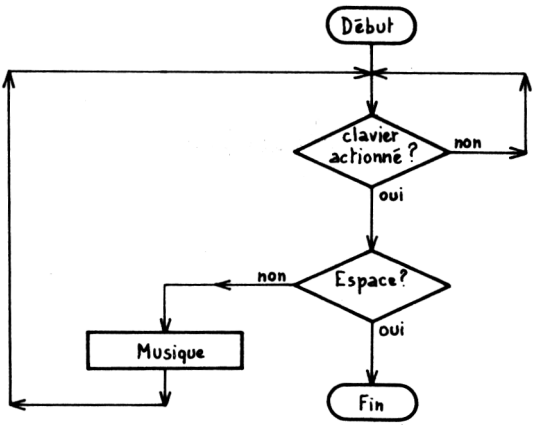
ORGANIGRAMME DU JEU

Le programme possède la structure hiérarchique à deux niveaux définie en annexe.

Le module de traitement consiste en :

- attente tant que la clavier n'est pas actionné ;
- test si espace ;
- musique.

L'organigramme du traitement est donc le suivant :



STRUCTURE DU PROGRAMME

Niveau 1 - **lignes 10 à 80** : Module de commande.

Niveau 2 - **lignes 1000 à 1980** : Affichage des notes et de leur correspondant-touche.
lignes 2000 à 2150 : MO5/TO7 joue.
lignes 3000 à 3080 : Fin du programme.

IDENTIFICATION DES VARIABLES

B : valeur numérique de la touche actionnée.

B$: touche-note activée.

I : index de boucles.

T$: tableau des notes.

```
10 :    REM PIANO
20 :
30 :      GOSUB 1000 'Affichage des notes
40 :      GOSUB 2000 'Piano
50 :      GOSUB 3000 'Fin du jeu
60 :
70 :    END
80 :    REM ****************************************
1000 : REM Affichage des notes et de leur correcPondan
t-touche
1040 :
1080 :    SCREEN 6,0,0 : CLS : LOCATE 0,1,0
1120 :    COLOR 3
1160 :    PRINT"Note    : do do# re re# mi fa fa#"
1200 :    COLOR 1
1240 :    PRINT"Touche :  Q  Z   S  E   D  F  T"
1280 :    COLOR 3 : PRINT
1320 :    PRINT"Note    : sol sol# la sib si do"
1360 :    COLOR 1
1400 :    PRINT"Touche :  G   Y    H  U   J  K"
1440 :
1480 :    LOCATE 0,12 : COLOR 6
1520 :    PRINT"APPuyez sur ESPACE Pour arreter."
1540 :    LOCATE 0,14
1560 :    PRINT"Pour jouer, aPPuyez sur les touches"
1570 :    PRINT"corresPondant aux notes desirees."
1600 :
1640 :    REM Lecture des notes
1680 :
1720 :      DIM T$(26)
1760 :      FOR I=1 TO 26
1800 :        READ T$(I)
1840 :      NEXT I
1880 :
1920 :      DATA ,,,MI,RE#,FA,SO,LA,,SI,O5D004,,,,,DO,
,RE,FA#,SIb,,,,SO#,DO#
1940 :
1960 : RETURN
1980 : REM ****************************************
2000 : REM Piano
2010 :
2020 :    B$=INPUT$(1)
2030 :    B=ASC(B$)-64
2040 :
2050 :    IF B<1 OR B>26 THEN B=1
2060 :    IF B$=" " THEN GOTO 2140
2070 :
2080 :    PLAY T$(B)
```

```
2090 :
2100 :    GOTO 2000 'Boucle de jeu
2110 :
2140 : RETURN
2150 : REM ****************************************
3000 : REM Fin du jeu
3010 :
3020 :    CLS
3030 :    COLOR 3
3040 :    LOCATE 5,12,1
3050 :    PRINT"Au revoir et a bientot..."
3060 :
3070 :    COLOR 6
3080 : RETURN
```

Thèmes célèbres

PRESENTATION

Cet utilitaire musical montre comment stocker des morceaux de musique sur support magnétique, et les interpréter par la suite.

Cinq morceaux sont proposés :

- pop corn ;
- histoires sans paroles ;
- classique ;
- Midnight express ;
- la truite de Shubert.

Un seul ordre Basic commande le générateur sonore. C'est l'ordre "PLAY".

Il est utilisé sous la forme PLAY A$ où A$ est une chaîne de caractères.

Cette chaîne peut être composée :

- de notes suivies éventuellement de # ou b.
- d'arguments.

Il y a cinq arguments possibles :

- **octave** : 01..05 où 01 est l'octave la plus aiguë,
  05 est l'octave la plus grave,
  04 est l'octave standard.
- **silence** : P Silence de la longueur d'émission d'une note.
- **longueur** : L1..L96 où L1 est la plus courte,
  L96 est la plus longue,
  L24 est la longueur standard.

- **tempo** (ou nombre de notes émises/minute)
  T1..T255 où T1 est le plus rapide,
  T255 est le plus lent,
  T5 est le temps standard.
- **attaque** : A∅..A255 où A∅ est une note continue,
  A255 est la note la plus amortie,
  A∅ est l'attaque standard.

ORGANIGRAMME DU PROGRAMME

Le programme possède la structure linéaire définie en annexe. Il se décompose en deux parties bien distinctes :

- programme musical (dans lequel l'ordre PLAY est activé) ;
- données correspondant à l'un des morceaux.

STRUCTURE DU PROGRAMME

lignes 10 à 420 : Programme musical.
lignes >430 : Données.

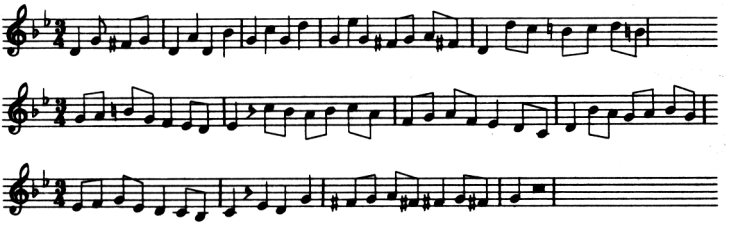
IDENTIFICATION DES VARIABLES

A$: chaîne d'activation de PLAY.

NP : nombre de PLAY consécutifs.

Si vous voulez entrer d'autres morceaux que ceux présentés ici :

- si vous ne possédez pas la partition du morceau, la première chose à faire est d'en établir une.
- l'écriture du morceau en langage MO5/TO7 est ensuite une chose enfantine. A titre d'exemple, comparez les données correspondant au morceau "Classique" et la partition du morceau "Classique" :



```
10 : REM THEMES CELEBRES
20 :
30 :   REM Titre
40 :
50 :     SCREEN 1,0,0 : LOCATE 1,1,0
60 :     ATTRB 1,1 : CLS : PRINT
70 :     PRINT"  THEMES CELEBRES"
80 :
90 :   REM Menu
100 :
110 :    LOCATE 0,8 : ATTRB 0,0 : COLOR 6
120 :    PRINT"    Je Peux Jouer :":PRINT
130 :    PRINT"     -> (1) PoP Corn"
140 :    PRINT"     -> (2) Histoires sans Paroles"
150 :    PRINT"     -> (3) Classique"
160 :    PRINT"     -> (4) Midnight exPress"
170 :    PRINT"     -> (5) La truite de Shubert."
180 :
190 :  REM Choix
200 :
210 :    LOCATE 0,20,1 : COLOR 3
220 :    INPUT"Votre choix (1,2,3,4,5 ou E)nd ";C$
230 :
235 :    IF C$="E" THEN 860 'Fin
240 :    NP=0 : A=VAL(C$)
241 :    IF A=2 THEN NP=74
242 :    IF A=3 THEN NP=151
243 :    IF A=4 THEN NP=224
244 :    IF A=5 THEN NP=259
245 :
250 :    REM Lecture des donnees non signifiCatives
260 :
265 :    IF NP=0 THEN 310
270 :    FOR I=1 TO NP+1
280 :      READ A$
290 :    NEXT I
300 :
310 :    REM Activation du morceau demande
320 :
330 :      READ A$
340 :      FOR I=1 TO VAL(A$)
350 :        READ A$
360 :        PLAY A$
370 :      NEXT I
380 :
390 :    RESTORE 'On Pointe sur la 1ere donnee
395 :    CLS
400 :
```

```
410 :     IF C$<>"F" THEN 10 'Boucle de jeu
420 :
430 : REM PoP corn
440 :
450 :    DATA 74,04L24MI,RE,MI,03SI,SO,SI,MI04P,MI,RE,M
I,03SI,SO,SI,MI04P,MI,FA#
460 :    DATA SO,FA#,SO,MI,FA#,MI,FA#,RE,MI,RE,MI,DO,MI
P,L48SO,MI,03SI04
465 :    DATA L24PSO,LA,L48SI
470 :    DATA LA,SOP,SO,MI,03SI04L24P,SO,LA,L48SI,LA,SO
L24P,MI,RE,MI,03SI,SO
475 :    DATA SI,MI04P,MI
480 :    DATA RE,MI,03SI,SO,SI,MI04P,MI,FA#,SO,FA#,SO,M
I,FA#,MI,FA#,RE,MI,RE,MI
490 :    DATA DO,MI
500 :
510 : REM Histoires sans Parlole
520 :
530 :    DATA 76,05L12MI,DO,04SO,MI,DO,MI,SO,MI,SO,L24L
AL12
540 :    DATA MI,SOP,05MI,DO,04SO,MI,DO,MI,SO,MI
550 :    DATA SO,05D004,SI,05RE,L24DOL12P,MI,L24SOL12,M
I,DOP,MI
560 :    DATA DO,04L248OL12,LAP,05MI,L24SOL12,MI,DOP,MI
,DO,04SO
570 :    DATA 05DO,D004,SI,05DOP
580 :    DATA MI,DO,04SO,MI,DO,MI,SO,MI,SO,L24LAL12
590 :    DATA MI,SOP,05MI,DO,04SO,MI,DO,MI,SO,MI
600 :    DATA SO,05D004,SI,05RE,D004P,L6SOP,SO,SO,L12SO
#,SOP
610 :    DATA SO,05DO
620 :
630 : REM Classique
640 :
650 :    DATA 72,04L24RE,L12SO,FA#,SOPL24,RE,LA,RE,SIb,
SO,05D004,SO,05RE04,SO
660 :    DATA 05MIb04,SO,L12FA#,SO,LA,FA#,L24REL12P,05R
E,DO,04SI
670 :    DATA 05DO,RE,04SI,SO,LA,SI,SO,L24FA,L12MIb,RE,
MIbP
680 :    DATA 05DO,04SIb,LA,SIb,05D004,LAFA,SO,LA,FA,L2
4MIbL12,RE,DO,REP
690 :    DATA SIb,LA,SO,LA,SIb,SO,MIb,FA,SO,MIb,L24REL1
2,DO,03SI04,DOP
700 :    DATA L24MIb,RE,SO,L12FA#,SO,LA,FA#,L24FA#,L12S
O,FA#,L24SO
710 :
720 : REM Midnight express
```

```
730 :
740 :    DATA 34,05L24RE,D0,04S0,MIb,RE,L36D0L24P
750 :    DATA 05RE,D004,S0,MIb,RE,D0,RE,MIb
760 :    DATA L48FA,L24S0,L96REL24,05D004,SIb
770 :    DATA LAb,MIb,FA,S0,LAb,SIb,L4805D004
780 :    DATA L24PLAb,SIb,05D0,RE,MIb,FA,S0,L96D0
790 :
800 : REM La truite de Shubert
810 :
820 :    DATA 31,04L24D0,FA,FA,LA,LA,L48FA,D0,L12S0,FA,
MI,RE,L48D0
830 :    DATA L12S0,FA,MI,RE,L48D0L24,D0,FA,FA,LA,LA,L4
8FAL24,D0
840 :    DATA FA,MI,L12RE,MIL24,FA,03SI04,D0
850 :
860 : END
```

Titi

PRESENTATION

Ce programme réalise le dessin de TITI sur une imprimante 80 colonnes en basse résolution, en utilisant les lettres de l'alphabet.

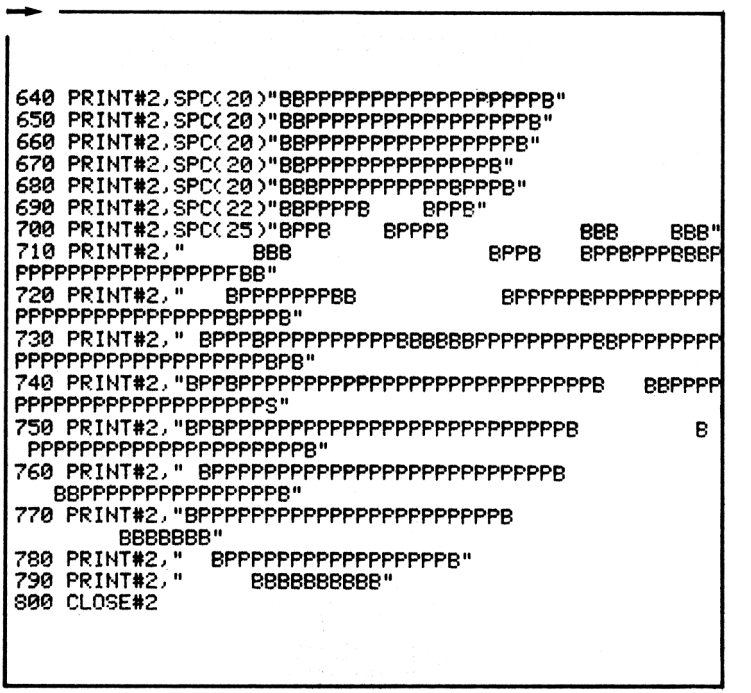
Plusieurs lettres sont utilisées, afin d'obtenir différents contrastes.

ORGANIGRAMME DU PROGRAMME

Le programme possède la structure linéaire définie en annexe. Il consiste en l'impression du dessin, ligne par ligne.

```
10 REM TITI
20 :
30 OPEN"O",#2,"LPRT:(80)"
40 PRINT#2,SPC(36)"O"
50 PRINT#2,SPC(30)"O    O"
60 PRINT#2,SPC(26)"O   O    O"
70 PRINT#2,SPC(27)"O   O    O"
80 PRINT#2,SPC(27)"O   O    O"
90 PRINT#2,SPC(24)"BPPPPPPPPPPPPPPB"
100 PRINT#2,SPC(19)"BPPPPPPPPPPPPPPPPPPPPPPPPB"
110 PRINT#2,SPC(16)"BPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
120 PRINT#2,SPC(13)"BPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
130 PRINT#2,SPC(11)"BPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
140 PRINT#2,SPC(9)"BPPPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
150 PRINT#2,SPC(8)"BPPPBPPPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
160 PRINT#2,"       BPPBPPPBPPPPPPPPPP PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
170 PRINT#2,"      BPPBPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
180 PRINT#2,"     BPPPPPBPPPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
190 PRINT#2,"    BPPPPPBPPBPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPBP"
200 PRINT#2,"    PPPPPPPPPPBPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
210 PRINT#2,"   BPPPPPPPPBPBPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
220 PRINT#2,"   BPPPPPPPPPBPPBPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
230 PRINT#2,"   BPPPPPPPPPBPPBPPBPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPB"
240 PRINT#2,"   BPPPPPPPPPPBPPB    PPPPPPPPPPPPPPPPPPPPP    BPPBPPPPPPPPPPB"
250 PRINT#2,"    PPPPPPPPPPB         PPPPPPPPPPPPPPPP      BPPPPPPPPPPPP"
260 PRINT#2,"    BPPPPPPPPPP         PPPPPPPPPPPPPPP       PPPPPPPPPPPPB"
270 PRINT#2,"     BPPPPPPPPP         PPPPPPPPPPPPPP      PPPPPPPPPPPS"
280 PRINT#2,"      BPPPPPPPP          PPPPPPPPPPPPP       PPPPPPPPPPB"
290 PRINT#2,"       BPPPPPPPP           PPPPPPPPPPPP        PPPPPPPB"
```

```
300 PRINT#2,"          BPPPPPPPP     OO     PPPPPPPPPPP
OO    PPPPPPPPB"
310 PRINT#2,"           BPPPPPP    0000    PPPPPPPPPPP    0
000    PPPPPPPB"
320 PRINT#2,"            BPPPPP    00000   PPPPPPPPPPP   00
000    PPPPPPB"
330 PRINT#2,"             BPPPPP   00000   PPPPPPPPPPP   0B
B00 PPPPPPB"
340 PRINT#2,"             BPPPPP   00000   PPPPPPPPPPP   0B
B00 PPPPPPB"
350 PRINT#2,"              BPPPPP 00BB0   PPPPPPPPPPP   0B
B00 PPPPPPB"
360 PRINT#2,"               BPPPP 00BBB   PPPPPPPPPPP   BB
B00 PPPPPB"
370 PRINT#2,"                BPPPPP 00BB PPPPPPPPPPPPPBB0
0 PPPPPPB"
380 PRINT#2,"                BPPPPPPPPP0BPPPPPPPPPPPPPPB00
PPPPPPPPB"
390 PRINT#2,"               BPPPPPPPPPPPPPPPP          PPPP
PPPPPPPPPPB"
400 PRINT#2,"               BPPPPPPPPPPPPPPBPPPPPPPPPPPPPBP
PPPPPPPPPPB"
410 PRINT#2,"               BPPPPPPPPPPPPBPP             PB
PPPPPPPPPPB"
420 PRINT#2,"                 BPPPPPPPPPBPPPPP        PPPPP
BPPPPPPPPS"
430 PRINT#2,"                    BPPPPPPPPPPPPPPPPPPPPPPP
PPPPPPB"
440 PRINT#2,SPC(26)"BDPPPPPPPDB"
450 PRINT#2,SPC(27)"BPPPPPPPB"
460 PRINT#2,SPC(28)"BPPPPPB"
470 PRINT#2,SPC(29)"PPPPP"
480 PRINT#2,SPC(29)"BPPPB"
490 PRINT#2,SPC(29)"BPPPB"
500 PRINT#2,SPC(26)"BBPBPPPBB"
510 PRINT#2,SPC(25)"BPPPBPPPPBB"
520 PRINT#2,SPC(25)"BPPPPPPPPPB"
530 PRINT#2,SPC(24)"BPPPPPPPPPPBB"
540 PRINT#2,SPC(23)"BPPBPPPPPPPPBPB"
550 PRINT#2,SPC(22)"BPPPBPPPPPPPPPBPPB"
560 PRINT#2,SPC(21)"BPPPPPPPPPPPPPPBPPPB"
570 PRINT#2,SPC(20)"BPPPPBPPPPPPPPPPPBBPPB"
580 PRINT#2,SPC(21)"BPPPPBPPPPPPPPPPPPBPB"
590 PRINT#2,SPC(21)"BPPPBPPPPPPPPPPPPPPB"
600 PRINT#2,SPC(21)"BPPBPPPPPPPPPPPPPPPB"
610 PRINT#2,SPC(22)"BBPPPPPPPPPPPPPPPPB"
620 PRINT#2,SPC(22)"BPPPPPPPPPPPPPPPPPB"
630 PRINT#2,SPC(21)"BBPPPPPPPPPPPPPPPPPB"
```



```
640 PRINT#2,SPC(20)"BBPPPPPPPPPPPPPPPPPPB"
650 PRINT#2,SPC(20)"BBPPPPPPPPPPPPPPPPPPB"
660 PRINT#2,SPC(20)"BBPPPPPPPPPPPPPPPPPB"
670 PRINT#2,SPC(20)"BBPPPPPPPPPPPPPPPPB"
680 PRINT#2,SPC(20)"BBBPPPPPPPPPPPBPPPB"
690 PRINT#2,SPC(22)"BBPPPPB    BPPB"
700 PRINT#2,SPC(25)"BPPB    BPPPB          BBB    BBB"
710 PRINT#2,"     BBB                BPPB   BPPBPPPBBBP
PPPPPPPPPPPPPPPPFBB"
720 PRINT#2,"   BPPPPPPPBB            BPPPPPBPPPPPPPPPP
PPPPPPPPPPPPPPPPBPPPB"
730 PRINT#2," BPPPBPPPPPPPPPPBBBBBBPPPPPPPPPBBPPPPPPPP
PPPPPPPPPPPPPPPPPPPBPB"
740 PRINT#2,"BPPBPPPPPPPPPPPPPPPPPPPPPPPPPB   BBPPPP
PPPPPPPPPPPPPPPPPPS"
750 PRINT#2,"BPBPPPPPPPPPPPPPPPPPPPPPPPPPB         B
 PPPPPPPPPPPPPPPPPPPPB"
760 PRINT#2," BPPPPPPPPPPPPPPPPPPPPPPPPPB
   BBPPPPPPPPPPPPPPPB"
770 PRINT#2,"BPPPPPPPPPPPPPPPPPPPPPPB
       BBBBBBB"
780 PRINT#2,"  BPPPPPPPPPPPPPPPPPB"
790 PRINT#2,"     BBBBBBBBBB"
800 CLOSE#2
```

Annexe

Cet annexe décrit le type de programmation employé dans ce manuel. Avant d'entrer dans le vif du sujet, posons-nous la question suivante : **qu'est-ce qu'un programme ?**

Voici une réponse qui ne se veut pas exhaustive : *c'est un ensemble d'instructions ou mots-clé stockés en mémoire, et qui ont pour but d'activer les ressources matérielles de l'ordinateur.*

Si ces instructions sont mises bout à bout, le programme risque d'être incompréhensible. Il est alors comparable à du texte en prose excluant ponctuation, paragraphes et chapitres. Cela n'est pas grave si le programme est simple. Si le programme est complexe, il devient vite impossible à déchiffrer.

Il apparaît donc que, comme en littérature, un programme manipulant un grand nombre de concepts doit comporter ponctuation, paragraphes et chapitres.

En informatique, la ponctuation porte le nom de **structure**, les paragraphes sont appelés **modules**, et les chapitres **niveaux.**

Comme en littérature, un programme comporte :

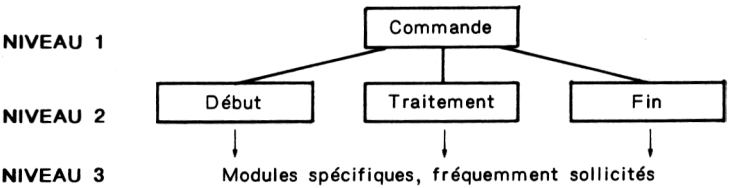
- plusieurs niveaux ;
- chaque niveau contient un ou plusieurs module(s),
- à l'intérieur d'un module, la strucutre sépare et organise les concepts manipulés.

Dans ce livre, les programmes sont relativement simples. Ils comportent une structure à deux ou trois niveaux :

- **le niveau 1** est constitué d'un seul module appelé module de commande ;

- **le niveau 2** se décompose en trois modules appelés début, traitement et fin ;
- **le niveau 3** s'il existe, se décompose en divers modules réalisant des tâches simples, fréquemment sollicitées.

Le schéma suivant, ou graphe hiérarchique, détaille cette structure :



Le module de début réalise la présentation et l'initialisation des variables et constantes du programme.

Le module de traitement constitue le corps du programme. C'est lui qui contient les principaux concepts manipulés pour réaliser le programme.

Le module de fin présente le résultat final, les remarques ou commentaires, et déconnecte l'ordinateur du programme.

Les programmes les plus simples ne nécessitent pas de structure hiérarchique à deux ou trois niveaux. Ils possèdent seulement un niveau, et leur structure est dite linéaire.

En conclusion, plus un programme est complexe, et plus il est nécessaire qu'il soit structuré et possède des niveaux différents.

Dans la pratique, je ne puis que vous conseiller d'écrire vos programmes de la sorte, car cela facilite grandement la lisibilité (et donc le dépistage d'éventuelles erreurs) et la maintenance.

Imprimé en Belgique
sur les presses du Scorpion
D/1984/3548/10
N° d'édition 87109.015.1
I.S.B.N. 2.87109.015.7

MegaO POCHE

BOITE A OUTILS POUR MO5 et TO7/70 19 PROGRAMMES EN BASIC

Qui n'a jamais regretté de ne pas avoir sous la main le tournevis ou la pince pour faire tel ou tel bricolage? Les ordinateurs par leur aspect universel sont comme les boîtes à outils (et les auberges espagnoles...): on y trouve ce que l'on y met. Cette collection MegaO-Poche n'a pas la prétention de livrer le secret des grandes applications informatiques, mais bien de mettre à la disposition des utilisateurs des petits programmes tout prêts qui leur permettront de résoudre de nombreux problèmes de la vie quotidienne.

35,00 FF

ISBN 2.87109.015.7

DIFFUSÉ PAR P.S.I.

Couverture : création et dessin Françoise Guillot, exécution Agraph